THE USE OF SYSTEMS ENGINEERING PROCESSES AND TOOLS TO DEVELOP A SYSTEM DYNAMIC SIMULATION MODEL OF ENGINEERING SUPPORT DURING THE DEVELOPMENT PHASE OF AN ACQUISITION PROGRAM

THESIS

Jason E. Bartolomei, First Lieutenant, USAF

AFIT/GSE/ENY/01M-01

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GSE/ENY/01M-01

AFIT/GSE/ENY/01M-01

THE USE OF SYSTEMS ENGINEERING PROCESSES AND TOOLS TO DEVELOP A SYSTEM DYNAMIC SIMULATION MODEL OF ENGINEERING SUPPORT DURING THE DEVELOPMENT PHASE OF THE ACQUISITION PROCESS

THESIS

Presented to the Faculty

Department of Aeronautical Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

in Partial Fulfillment of the Requirements for the

Degree of Master of Science in Systems Engineering

Jason E. Bartolomei, BS

First Lieutenant, USAF

March 2001

AFIT/GSE/ENY/01M-01

THE USE OF SYSTEMS ENGINEERING PROCESSES AND TOOLS TO DEVELOP A SYSTEM DYNAMIC SIMULATION MODEL OF ENGINEERING SUPPORT DURING THE DEVELOPMENT PHASE OF THE ACQUISITION PROCESS

Jason E. Bartolomei, B.S.
First Lieutenant, USAF

Approved:

E. Price Smith — 8 MAR 01
E. Price Smith (Chairman) — date

Curtis J. Spenny — 8 Mar 01
Curtis Spenny (Member) — date

Patrick J. Sweeney — 3/7/01
Patrick Sweeney (Member) — date

ACKNOWLEDGMENTS

This thesis is dedicated to my wife, whose love, encouragement and support enabled me to accomplish the impossible. I am truly one of the most blessed men in the world.

A special thanks also goes to:

My sons: For loving me, regardless of what kind of grades I get. A man could not ask for better sons.

Lt. Col. E. Price Smith: Your superb mix of freedom and thoughtful direction ensured my success.

Dr. Curtis Spenny: For sparking my systems interest.

Dr. Pat Sweeney: If I can encourage others as much as you have encouraged me, the world will be a better place.

Dr. Mike Shelley: You asked all the right questions. Thanks for you prayers, insight, and time.

Mr. Gary Adams: Your keen insight and understanding amazes me. No person has taught me more about systems engineering and thinking.

Mr. Gerry Freisthler: Thank you for your direction and support.

Mr. Jon S. Ogg: You have taught me two lessons that I will never forget: 1.) It is better to do a few things well, than many things half-heartedly. 2.) You don't get exceptional work out of people by expecting or accepting mediocrity. Without your professional patience and personal direction, I would have not been able to complete this thesis.

Jason E. Bartolomei

PREFACE

This thesis reports on the development of a systems dynamics model of the engineering function of an Air Force development activity and proposes a generic methodology to approach the problem. I recognize that there are several categories of people who may read this thesis, and I wish to provide a general guide for the various audiences.

For those who are interested in a summary of the system research should read chapters one and four first, and then turn to chapters two and three if further detail is desired. Those who are interested in the model as a foundation of further research should focus their attention on Chapters two and three. Individuals who are unfamiliar with the system dynamics symbols should scan Appendix A before reading chapter three.

I hope this guide will save the reader time in gaining the degree of understanding desired.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Abstract

Due to the increase of system complexity and the existing draw down of manpower allocations, today's acquisitions environment desperately needs a systems approach to decision making. Many studies have been performed to model the entire government acquisition environment. Due to the high degree of aggregation, front line decision-makers have had no use for the information these models provide.

This research focuses on the Air Force's largest functional support element in aircraft systems development, engineering. I will only consider one phase of the government acquisition cycle the Engineering, Manufacturing, and Development (EMD). This is the development cycle, which begins with initial contract award (Milestone II), through the production approval (Milestone III). The structure of this model will be a building block to help USAF leadership in the determination of required engineering skill-set and manpower to perform activities which can meet short term requirements while minimizing the intrinsic cost, schedule, and performance risks associated system development. The simulation model will be used by USAF leadership as an alternative decision making tool for manpower allocations for government organic engineering workforce during an eight year development effort.

In addition, this study investigates the benefit of using system engineering tools and processes, like Functional Allocation (FAST) and Quality Functional Deployment, to improve the process for generating system dynamics simulation models.

For years, the systems engineering field has developed tools to graphically represent complex system structure. Graphical representations allow individuals and teams to visually identify interrelationships and dependencies within a system. Academic research and the successful implementation of these tools within the industrial communities validate the utility of

these tools. These tools include Unified Program Planning, Quality Functional Deployment, House of Quality, and Design Structure Matrix. (Blanchard and Fabrycky:1999)

This thesis presents a new tool, Management Causal Matrix, for system dynamics modeling community. The matrix is very similar to the more traditional systems engineering tools, yet has been customized for the systems dynamacist to highlight system interdependencies and organize the causal structure for a management system

A system dynamic simulator was developed to examine government engineering resource allocation during the development phase of an acquisition program. By using systems engineering approach, the scope of the previously poorly understood system was efficiently determined and a dynamic model was produced.

THE USE OF SYSTEMS ENGINEERING PROCESSES AND TOOLS TO DEVELOP A SYSTEM DYNAMIC SIMULATION MODEL OF ENGINEERING SUPPORT DURING THE DEVELOPMENT PHASE OF THE ACQUISITION PROCESS

1. Introduction

Background

Since end of the Cold War, the U.S. Government has made concerted efforts to reduce the costs of its acquisition processes. One strategy has been the elimination of activities that are not necessary or cost effective. This has been a difficult transition, as the entire Defense industry wrestles with conducting business within a reformed acquisition environment. No government institution has felt the ramification of the new business practices as much as the System Program Offices (SPOs). In the USAF, the SPO is responsible for managing the entire lifecycle of a designated system. This includes the initial concept exploration and requirements generation through the development, production, and retirement of the system.

For years, SPOs participated in many activities that overlapped contractor efforts. The generation of a Government Statement of Work, Government specifications, and other standard practices made government acquisitions very costly and inefficient. SPOs were filled with military and government service engineers, financial managers, program managers, and contract specialists. All were assigned to duplicate contractor effort in managing the cost, schedule, and performance of the system.

In 1991, the Government Acquisition community began to rethink the manning and skills mix for weapon system development efforts. AFSC (Air Force Systems Command) Commander

General Yates and the Acquisition leadership were faced with a problem. General Yates stated, "AFSC does not have a credible, quantitative process to determine the right numbers and skills of people to properly match acquisition workload, nor can it effectively defend the acquisition workforce" (Yates:1991).

A team embarked on a three-year project to determine a process for managing the existing acquisition force. The strategy to attack the problem was first to utilize other government manpower models in combination with "top-level intellectual involvement" to develop USAF-specific manpower models. The commander believed that it was necessary to determine, quantify, and defend the value of an organic acquisition workforce; otherwise, the acquisition community would continue to dissolve. Although the effort received great visibility and attention, the team was unable to determine manning models that could defend the value of maintaining an organic workforce.

In 1995, the Secretary of Defense (SECDEF) mandated several changes to the government acquisition process. The Air Force translated these mandates into several "Lightning Bolt Initiatives" released by the Office of the Assistant Secretary of the Air Force for Acquisitions (SAF/AQ) to facilitate a reform process due to the dismantling of the Defense budgets. Lightning bolt number 3 was entitled, "System Program Office 'Slim-Fast' Plan." The goal of the Lightning bolt was to "eliminate all unnecessary SPO activities thereby reducing program costs" (Druyun:1995).

The methodology for attacking the problem was to investigate programs that had demonstrated success in a reform-like environment. One arena was classified acquisitions, or Special Access Required (SAR) programs. Historically, SAR programs had successfully performed their mission with a substantially smaller workforce than unclassified SPOs

(Druyun:1995). A tiger team assembled to develop tenets applicable to unclassified programs which could help SPO Directors (SPD) to achieve efficiencies in SPO operations and ultimate reductions in manpower.

New SPO manning goals were established setting new thresholds for manning acquisition programs. Large development programs which previously employed over 300 military and civil servants were limited to 140 total personnel. Systems in the production phase that once had 150 employees were challenged to employ 50. The goals represented the total workforce manning, including government and support contractor resources. These goals were quite aggressive, up to a 90 percent reduction for some efforts. No apparent analysis was presented to verify or validate these numbers.

After two years of interviews, assessments, and analysis of streamlined acquisition practices, the Lightning Bolt team presented a list of "inherent government functions" to be retained by a SPO. These functions include the following:

I. Contracting - The processing of the contractual documentation and obligation of the government to pay for provided materiel and/or services.
II. Program Management - Evolves around understanding and managing risk through evaluation of program cost, schedule, and performance. A key component of this area is technical assessment - determining pass/fail of intermediate and final test criteria.
III. Requirements Determination - Setting the quality, quantity, and performance characteristics required from a procurement.
IV. Budgeting and Financial Management - The programming for, obligation of, and accounting for SPO funds.

It was determined that all other activities could be "source through the prime contractor, support contractors, or eliminated completely" (Lightning Bolt #3). The functions were then reduced to several tenets described by the team as approaches which, if implemented intelligently, would lead to streamlining improvements. These tenets are as follows:

A. Aggressive Risk Management - A move away from risk avoidance toward risk management
B. Insight vs. Oversight - Understanding contractor processes and managing the program via process metrics
C. Clear Accountability in Design (CAID) - To the extent practical, the Air Force assumes no design responsibility below the functional baseline (system specification) level until the end of EMD

D. Integrated Weapon System Management (IWSM) - Non adversarial team membership to include the contractor and user
E. Reduce Contract Data Requirements List (CDRLs) - Use existing contractor systems for insight
F. Communication of Performance specifications - What we want the system to do, not how to build it
G. Flat SPO structure - accessibility to the Program Director
H. Maximum use of electronic media
I. Maximum use of commercial off the shelf (COTS) items—Only develop what needs to be developed.

Additional tenets were provided to the government system program director (SPD) to consider. These embrace the following key elements:

A. Maximize the teaming efforts to include other government agencies
B. Establish long-term government-contractor relationships
C. Minimize the number of contracts/line items and make them milestone based
D. Contractor develops a logistics support plan focused on 4 critical parameters
E. Minimize and refocus engineering staff
F. Borrow expertise rather than maintain within the SPO

Openly, the team admitted that the "report is not a 'cookbook' or a mathematical model for SPO sizing." They explained that their intent was to present a "tool box" for SPDs, which is to be "applied thoughtfully, based on the careful judgment of the program office personnel" (Lightning Bolt #3:1995).

The government acquisition community generally agreed that these steps could optimize certain portions of the acquisition cycle. However, many of the tenets were difficult to integrate in existing programs and some were unrealistic for some programs.

It was interesting that the tenets removed the engineering function. Government engineering support was one of the largest resources previously utilized in the system programs offices. Removing engineers would immediately reduce the manpower load by 50 percent for development activities. This may be attractive for the acquisition reform leadership, but it was unreasonable for the services and the contractors who rely on the engineering support to provide independent assessment of a program's technical performance. Many philosophical debates persisted throughout the acquisition community on the usefulness of maintaining a large

engineering function within the SPOs. However, no analytical solutions could quantify the value of organic engineering support.

System Dynamics Modeling of Government Acquisition Systems

In the Air Force, I located five efforts at modeling all or part of the DoD acquisition system that were performed in the late 1970's and early 1980's. These include Elder and Nixon, Lawson and Osterhaus, Kaffenberger and Martin, Whittenberg and Woodruff, and Gonzalez and Sarama. The first 5 studies were performed during the Cold War arms race. Whispers of U.S. Government acquisition reform were evident in the text, but there were irreconcilable differences in fundamental assumptions, model boundaries, political, and environmental issues to further their research.

Elder and Nixon developed a conceptual model of the USAF program management activities being performed at Aeronautical Systems Division. They did not produce a completed model (Elder and Nixon:1976).

Lawson and Osterhaus developed a six-sector model depicting the entire DoD acquisition process, circa 1980. The causal relationships described in the text were not well understood or parameterized to be useful. The process was completed by a series of interviews with Senior Defense acquisition leaders and executives (Lawson and Osterhaus:1978).

Kaffenberger and Martin added four more sectors to Lawson and Osterhus' model. Their model captured and described several causal relationships and systemic behaviors of the US-USSR arms race. Their contribution rested primary on the structure of the system and provided no validation or analysis (Kaffenberger and Martin:1979).

Whittenberg and Woodruff provided a capstone thesis, which completed the work of the previous studies as well as the work of Bechtel and Sweeney. Their comprehensive model of the acquisitions was quite an undertaking. Like the others, the model was designed for, and aggregated at, the highest level. Though the model was run and the provided some useful information to acquisition policies for the cold war environment, little was applicable to today's current world environment (Whittenberg and Woodruff:1982).

Gonzalez and Sarama developed a causal model that looked at the acquisition process from a Washington DC policy perspective. They modeled sectors including the political, economic, contractor, and multiple government agencies at a very high level of aggregation. They were unable to develop a computerized model of the system, but the well-defined causal structure provided insight to the complexity of the system (Gonzalez and Sarama:1999).

The very high level of aggregation and the enormous scope of these projects provided little detail or information for frontline managers to make decisions.

Model Objectives

The general objective of this research was to develop a conceptual understanding of the complex, dynamic nature of a government development program, and subsequently develop a computerized model that reflects the structure of the engineering function within a system program office (SPO). In the USAF, the SPO is responsible for the managing the lifecycle for a designated system. The lifecycle includes the initial concept exploration and requirements generation through the retirement of the system. The SPO consists of US government employed program managers, engineers, financial managers, and contract specialists.

The specific objectives were as follows:

1. Identify the structure of the system using traditional systems engineering tools such as Functional Analysis Systems Technique and Design Structure Matrix in concert with a system dynamics approach.

2. Isolate the interactions and influence of the components and variables within the system.

3. Describe the decision structure that determines the allocation of engineers to the SPOs.

4. Construct a mathematical model that represents the components, relationships, information flows, and decision policies of the system.

5. Develop a computerized model that can be used as a learning laboratory for policy analysis and development to optimize engineering support of a high-risk USAF development activity.

6. Identify areas of sensitivity or critical issues in engineering manpower allocation.

Contributions to Research

In addition to developing a model to determine engineering manning levels in SPOs, this thesis also presents a new methodology for creating system dynamics simulation models. The methodology proposes the use of several traditional systems engineering tools and processes that are helpful to elicit knowledge, organize system structure, and define system boundaries and degree of aggregation.

Boundary Definition

Defining the scope of a system is one of the most difficult aspects of model development. Stafford Beer argues that scope is a major issue when using the scientific method to solve problems. He contends that many times erroneous results are produced when the boundaries of the system do not encompass the whole problem (Beer:1972).

System dynamacists have an array of tools used to map a systems structure. These tools include the Model Boundary Chart, Subsystem Diagram, Causal Loop Diagrams, Stock and Flow map, and Policy Structure Diagram (Sterman:2000). Modelers commonly employ some or all of these mapping tools in the formulating the dynamic hypothesis and system structure definition phase of simulation development. If employed properly, the tools greatly benefit both the modeler and the customer. The primary function of these tools is to visually organize the complex structure of the system and to communicate the system structure to others. Unfortunately, for large and complex systems some of the tools are less effective. Causal loop diagrams can grow to be enormous and difficult to follow. Complex subsystem diagrams often fail to provide all of the critical information needed to build a model and often it is cumbersome to translate the information into stock and flow structure.

The process to define the degree of aggregation and model boundary definition is one of the least understood aspects of model generation. Defining the scope of a new product, organization, or system and carefully understanding the many facets of a system is foundation systems engineering. System engineers and complex system designers use many methods, tools and processes to define the boundary and scope of complex systems. This thesis investigates to the use of traditional and modified systems engineering tools as alternative methods to define system boundaries.

Many of the systems engineering and design tools provide a visual architecture of a model. System modelers use subsystem diagrams to convey information on the boundary and the level of aggregation in the model by showing the number and type of different organizations or agents represented"(Sterman:2000).

To demonstrate the need of subsystem diagrams Sterman and Forrester both reference two subsystems of Forrester's corporate growth model. (Forrester:1962) In the development of the corporate growth model, Forrester presents a subsystem diagram to illustrate the information flows of orders, product, and money between the company and market he was attempting to model.

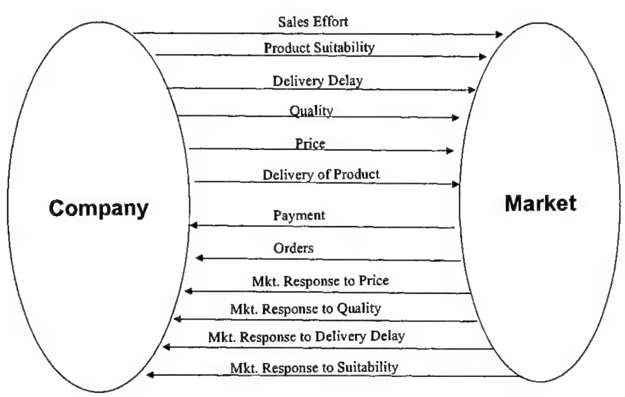


Figure 1.1. Subsystem Diagram

Forrester states: "Defining the system boundary and the degree of aggregation are the two most difficult steps in successful modeling. In this particular study, part-time effort for about two years was devoted to false starts before arriving at the point shown in Figure 1.1. Thereafter, only eight weeks were required to create the entire system." (Sterman:2000)

The subsystem diagram Forrester generated seems quite simple, it is hard to imagine that it would take 2-weeks, much less than 2 years to create the diagram. Expert knowledge, experience, and many other factors are required to successful develop even the simplest

boundary model. Though my literature searches it seems that eliciting the knowledge of the system structure is more of an art then a science. Many papers have been written on how to interview the experts of the system and describing the behavior to be modeled. But little has been written on the actual process to develop these structures.

System engineers and complex system designers use many methods, tools and processes to define the boundary and scope of complex systems for years. Defining the scope of a new product, organization, or system and carefully understanding the many facets of the system is foundation systems engineering.

Many formal tools and processes have evolved which allow systems engineers to carefully map the architecture of the system. Processes include the functional allocation process, systems architecting using tools like Quality Functional Deployment, House of Quality, Function Flow Bock Diagramming, House of Quality, Unified Program Planning, and Design Structure Matrix. These tools and their associated process enable the systems engineering and program management communities get a handle on the entire scope of a system (Blanchard and Fabrycky:1999).

Today, many of these tools and processes are being used to analyze traditionally above the design-room and shop floors systems, especially management systems. Consultants use traditional system engineering techniques to analyze management systems to help corporations lean-out, re-engineer, and optimize organizational structure. Unfortunately, the tools were designed to structure static systems (like automobiles, aircraft etc).

When consultants use these tools to investigate manpower allocation and organization structure, the tools do an excellent job at rigorously describing the functions the organizations perform and the resource allocations. However, They do not sufficiently describe the

interdependencies and interrelationships found within the system structure and fail to give management any insight the dynamic behavior of the organization environment. I believe there is mutual benefit in using the traditional system engineering tools with system dynamics simulation. Where system dynamics is weak on providing proven methods and tools for model boundary development, it is strong in modeling behavior of organizational management decisions. Countless management flight simulators have been successfully developed to simulate the management environment so managers can understand the ramification of decisions and policy (Sterman:2000). Using these methods, processes, and tools in concert could be greater than the sum of the parts.

2. Methodology:

The methodology is broken into several sections. The first section explains the use of a traditional systems engineering tool, Functional Analysis Systems Technique (F.A.S.T.), as method from group knowledge elicitation. The next section, describes a new process and matrix based tool to organize system structure. For readers interested in the system dynamics model, I suggest skipping to chapter three.

Information Gathering:

A military development program is a complex and poorly understood system of multiple interrelationships and interdependencies. The role of the engineering function within this structure was extremely difficult to define initially. Several group and individual knowledge elicitation techniques were used to gather the information. These methods and tools are described in the following sections.

<u>Functional Analysis System Technique (F.A.S.T.)</u>

Information gathering is the most important, and often the most difficult, phase of dynamic system model building. Many methods have been proposed to elicit knowledge from systems experts. Most modelers avoid the social and political barriers found in group elicitation. Instead, they focus on various interviewing techniques of individuals, as compared to groups. Properly employed, however, group elicitation can be an effective and extremely efficient method of attaining the necessary knowledge to analyze and model a system. This section discusses the use of the Functional Analysis Systems Technique (F.A.S.T.) diagramming method as a tool for eliciting knowledge during the information-gathering phase of the modeling process.

The Functional Analysis System Technique Process:

F.A.S.T. was invented during the value analysis (VA)/value engineering (VE) revolution of the 1960's. It is a rigorous method for understanding complex systems by converting the "activities" performed in a system to the "functions" performed by the system for its customers. System engineers and value analysis specialists use this method for product improvement, process improvement, systems design, and systems architecture. Its creation marks the completion of the formal information-gathering phase and defines the current state of a system at a high level.

For example, which is more important to understand from a systems perspective: the fact that today someone reviews a document (their activity) or the fact that accomplishing that activity improves security (their function) in the organization? While activity is important to getting a job done, it does not necessarily benefit the customer. In fact, function is what the customer ultimately pays for while activity is what they get. However, this activity can become narrow and self-serving. The VA specialists have a term for this: they call it selfish-sectional efficiency. It means the individual or unit becomes extremely efficient at the expense of the rest of the organization, and more importantly, at the expense of the customer.

A non-process example may help to illustrate why not understanding "function" can severely affect an organization. Gas-powered lawn mowers have been around for a long time. Yet, those highly skilled manufacturers, with big, fancy manufacturing plants missed one of the most important market niches of the last 40 years: the string trimmer. Why is this? It's because lawn mower people had tunnel vision within their everyday "activities"— making lawn mowers. They missed the broader consideration of the function they were providing for their customers: "groom property." Hence, the multimillion-dollar string trimmer market emerged. Unfortunately,

the lawn mower manufacturers and their customers didn't even care that it was not done with a lawn mower. In fact, it was a non-mower manufacturer who introduced the string trimmer and stole a good chunk of the business from mower companies. Even today, many mower manufacturers have still not gotten into the trimmer market because they are still stuck in the activity of mowing grass vs. the broader function of grooming property.

<u>The F-22 F.A.S.T. diagram Example</u>

This section will examine, using the F.A.S.T. exercise, understanding of a highly complex organization: a government weapon system program office (SPO) team. The specific program is the Air Force's F-22 Advanced Tactical Fighter SPO. Leadership from the F-22 SPO desired to use F.A.S.T. and other system analysis tools to improve their office and program management practices. As part of this effort, they decided to map their entire F-22 program using a F.A.S.T. methodology.

F.A.S.T. is a comprehensive, hierarchical block-diagramming tool that visually portrays the key functions an organization performs for its customers in a cause and consequence fashion. F.A.S.T. represents the current state of a system in functions without regard to timing and flow of activities. This raises the F.A.S.T. diagram from the normal activity-centered block-flow diagram to the higher-level function diagram. In value analysis, this change gives problem solving teams a new perspective on their situation. In turn, it allows them to be much more creative in later brainstorming sessions. For the modeler, F.A.S.T. is a highly efficient method to capture a detailed model of a system's functions.

<u>Use Organizational Experts and Work Offsite</u>

The key to generating a successful F.A.S.T. model is to ensure that the right people are chosen for the F.A.S.T. team. Organizational experts are the main contributors to a F.A.S.T.

diagram. Gathering F.A.S.T. data is best accomplished at a meeting held off site from the regular work environment and with the leadership of the organization. In this example, it was done with the senior members of the Air Force's F-22 SPO.

<u>Knowledgeable Participants of the F-22</u>

The F.A.S.T. team was led by the second in command, the deputy system program director. Other members included the chief engineer, deputy chief engineer, chief of the Contracting Division, chief of the Financial Division, chief of the weapon system program managers, flight test director, and lead weapon system engineer. The participants were selected for their knowledge of the overall system and for their respective decision-making roles in their groups within the SPO.

<u>Role of the Facilitator(s)</u>

Because of the size of the F.A.S.T. team, and due to inexperience with F.A.S.T. diagramming, three facilitators were selected to guide the team. Two were new to F.A.S.T. and a third party was a F.A.S.T. expert. The two new facilitators were trained in the F.A.S.T. process before the team began. The three then guided the team through the process.

<u>Customer Definition</u>

One of the principles of a quality design is to ensure all functions and activities are traceable to a customer requirement. So, before any functional allocations could be performed, customers and objectives had to be defined. As such, the first step in the F-22 process was for the team to identify the customers of their system.

For the F.A.S.T. exercise, customers were defined as any party that directly benefited from the products and services that the System Program Office generated. After a short discussion, the team quickly identified the main customers of the system—Air Combat

Command, the Pentagon (SAF/AQ), the AF Audit Agencies, internal SPO leadership, and the F-22 contractors.

Definition of Function

Since the primary objective of a F.A.S.T. diagram is to teach the team members to think in terms of higher–level functions rather than everyday activities, the team was assigned small samples of a system on which to practice. The functional approach to problem solving is the cornerstone of value analysis (VA) in that it translates the structure of any system into a structure of words. In short, synthesizing a system in terms of functions deepens the team's intuitive appreciation of the entire system (Fowler:1990).

Functions are reduced to simple, two-word, verb-noun descriptions of each activity. These F.A.S.T. functions can also be called a "functive" to minimize the confusion with organizational functions like engineering, finance, or contracting.

Below is an example of a functive describing an automobile control system:

- Drive car
    - Monitor environment
    - Monitor instrument data
    - Control direction
    - Control speed
    - Control visibility
    - Control human comfort
    - Control car health

Drive car is the higher-level function. The subordinate functions represent the means to achieve the higher level function. This example describes a hardware design. The functions defined for management systems are quite different.

Team Size

Studies have shown that teams of 5-6 people work well together (Fowler:1990). Above this number, teams tend to drift into small, informal groups with a core of only three to four people doing any real work. To head off this tendency, groups should be composed of five people. Problems can also arise when teams are comprised of less than five. Smaller groups lack sufficient understanding of the overall system to create an effective F.A.S.T. diagram.

Generating Functions

Once the team understood the concept of function, the facilitators led the team through an extensive brainstorming session—to create functions. Each sub-team was asked to create 50 –to 100 separate functions. As a procedure point, functions were first written on Post-it Notes® and then transcribed onto easel pads for all the team to see.

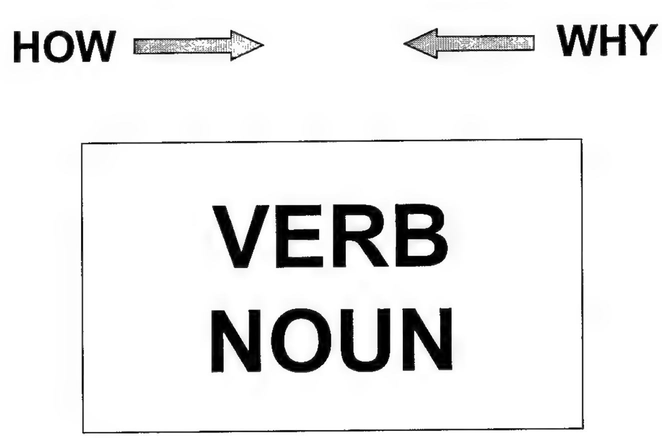


Figure 2.1 Function

One of the easiest ways to define functions is to ask "why" an activity is done. This questioning is not done to challenge someone's job; it is done to determine the reason the activity

exists—from the customer's perspective. Describing functions in two words, a verb-noun combination, is often difficult. This is because people tend to think in terms of the way we name things. For example, the Air Force has a coordination cycle on all documents before the boss can sign them. This means that every group that has an interest in the document has seen it in final form and either agrees or disagrees with it. Therefore, when the team was asked what the function of the coordination activity was—why coordinate—the response was varied. The first function suggested was "coordinate document." However, this merely describes the activity in two words; it is not a function. To get at the real function, the facilitator then asks "why" people coordinate on a document? At this point, the following real function(s) began to emerge: acknowledge read, verify content, signify agreement or disagreement, and authorize action. It is typical that one activity will generate several functions. However, once a function is defined, it does not need to be defined again if it occurs elsewhere in the system. Keep in mind, the F.A.S.T. diagram shows the state of the system over all time and, therefore, does not need to show duplicate or repetitive functions. Figure 2.2 is an example of a function.

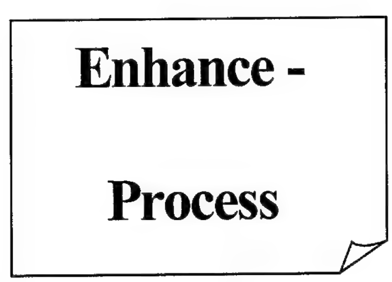


Figure 2.2 Two-word function description of the activity—file documents—on Post It Notes ®

Mapping Functions

Upon completion of the function generation phase, the team began the functional analysis by creating the F.A.S.T. diagram. Each of the functions remaining, after a scrub to delete repetitious functions, was compared to the other functions and was placed on the map. The functions were organized by comparing one to another using a "how and why" arrangement. A sample how-why relationship is shown below in Figure 2.3.

To test for the correct sequence between the Maintain Communications and Sustain Operations functions, put the two in a line and ask "why" of one to the right. In the example shown, ask "why maintain communication. If the answer—to sustain operations—makes sense, the functions are in the correct order. Test this by asking "how" in the opposite direction—how does one sustain operations, if the answer is by maintaining communication, then the two are in the correct order. The how-why logic also has a built-in test. To see this, switch the two functions and ask the same questions again. You will notice that the how-why questions no longer make sense. This shows the functions are in the wrong order. Repeat this how-why process for the next logical functions to the left and right of these two. In the end, a F.A.S.T. diagram will be created.

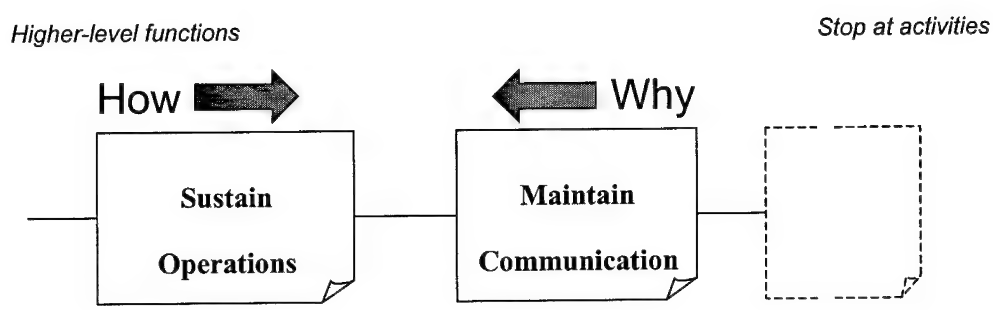


Figure 2.3 How –Why relationship verifies diagram

The final structure is a horizontal, logic diagram verified with how-why logic. Primary functions are left-most on the diagram and supporting functions extend to the right in descending

order of generality. The diagram stops on the right when further asking "how" could only be answered by putting "activities" on the diagram.

After several iterations of the mapping process, the team agreed on a final F.A.S.T. diagram that was representative of the all the key functions the system performed.

Ultimately, the F.A.S.T. exercise is a process of creating a functional flow block diagram of a system that already exists—its current state. The entire process mirrors what designers of complex system perform when creating a system's architecture. The entire F-22 F.A.S.T. is located in Appendix B.

Products and Services

Once the team developed the structure of the functions performed within the system, it then identified all of the key products and services performed in support of each function. In addition, the team identified the customers associated with each product and service and the party responsible for executing the tasks.

Allocation of Efforts

In the next step, the team determined the resource allocation for the support of each function. Experts representing each critical area identified the number of resources required to support each function. The team was then able to assign those resources to each function. This information allowed the team to locate areas of excess and make efforts to trim those areas.

Unfortunately, when the teams used F.A.S.T. as an analytic tool to make decisions, there was no consideration of the dynamic relationships between function and activities. The failure to consider these interdependencies limits the ultimate usefulness of the F.A.S.T. model as an independent tool for decision making. Though not effective as a stand-alone analytical tool, the

F.A.S.T. does provide a rigorous approach for identifying the functional architecture of the system and therefore has great value to the expert system designer. Other systems analysis tools, like system dynamics modeling, can enhance the benefit of F.A.S.T. models by capturing the interactions of the components of the system.

F.A.S.T. Summary:

Model builders can learn a great deal about an organization quickly by using this powerful diagramming tool. The F.A.S.T. diagramming exercise provides the system dynamics modeler multiple benefits. First, F.A.S.T. is an efficient method for defining a system's architecture. Second, completing the F.A.S.T. exercise allows the modeler and the customer greater insight into the system. Lastly, the F.A.S.T. model can enhance a modeler's credibility with the customer and can shorten the information-gathering phase for model development.

Management Causal Matrix:

A Management Causal Matrix enables the modeler to capture all of the critical interdependencies within a system using matrices to highlight causal relationships. The process in generating the information necessary to populate the MCM provides an intuitive method for eliciting the critical knowledge to create a system dynamics model.

The MCM is an adaptation of other matrix-based tools utilized by the quality and system engineering communities. The matrix structure is most similar to one of the first system engineering tools called Unified Program Planning.

Unfied Program Planning (UPP)

The purpose of Unified Program Planning (UPP) was to provide an intuitive method for graphically representing the interdependencies found within a system. As the grandfather of the

house of quality, UPP was created to display the multiple interrelationships that exist within a complex system. The common theme underlying system engineering tools and processes such as UPP remains fundamentally constant. Each tool is developed to provide a logical flow between customer requirements and the organization of elements that satisfy those requirements. Program managers, systems designers, and product developers commonly use these tools to define the scope of development. All elements within the system are traceable to the customer.

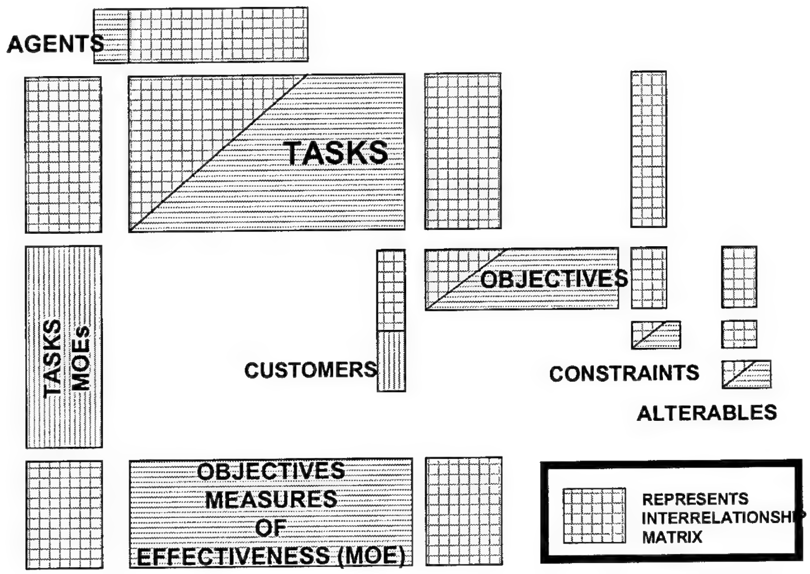


Figure 2.4 Unified Program Planning

Figure 2.4 is a simplified, UPP diagram. The critical elements of a system are defined on the diagram. Customers, system objectives, tasks to be performed, and measures of performance are all identified. The matrices are used to highlight interrelationships found within the system. The diagram is used by system engineers to ensure they have considered all of the critical elements within the system. Hill and Warfield's paper provides an excellent description of the UPP and gives a good example (Hill and Warfield:1972).

The original intent for UPP was to use it as a program-planning tool to address stakeholders' objectives, as well as the activities and constraints found within a system development effort. Although thorough and rigorous, it was a bit cumbersome to explain and somewhat difficult to understand. Therefore, I tailored the original flow and developed a new model, which I found was easier to understand and more conducive for the system dynamics modeling

MCM Methodology:

This section presents a methodology for constructing a MCM. The arrows in figure 2.5 show where the matrices are located and the flow of information feedback. The information used to create the MCM described in this section was elicited using the F.A.S.T. exercise described above.

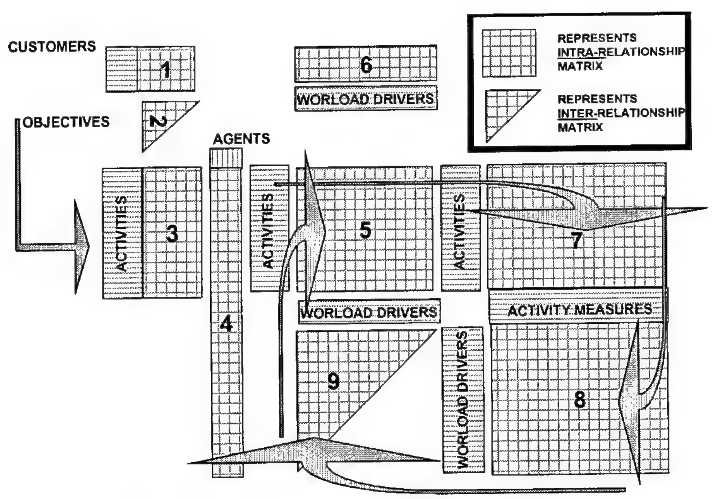


Figure 2.5 Management Causal Matrix

Matrix 1- Identify the customer:

The first step in creating the MCM is to identify all of the internal and external customers associated with a system. In the government acquisitions environment these customers are identified as the following:

SPO: the internal leadership of the management organization

KTR: Contractor, the commercial entity developing the weapon system.

PENT: The Pentagon

USER: The United States Air Force Major Command purchasing the weapon system

GAO: The Government Accounting Office

OBJECTIVES	Max. Manpower Effectiveness	Gather Information	Monitor Performance	Inform Customer	Obtain Funding	Negotiate Contract	Manage Risk	Ensure Executability

CUST.							
INT	X	X	X	X	X	X	X
KTR			X	X	X		X
PENT	X	X	X		X	X	X
USER	X	X	X			X	X

Figure 2.6 Matrix 1: Customers vs. Objectives

Matrix 2 - Define the system's objectives:

SPO/IPT OBJECTIVES							
				X	X	X	Ensure Executabillity
X			X			Manage Risk	
X			X		Negotiate Contract		
X			X	Obtain Funding			
X		X	Inform Customer				
X	X	Monitor Performance					
X	Gather Information						
Max.Manpower Effectiveness							

Figure 2.7
Matrix 2: Objectives Intra-relational matrix

The next step in developing the MCM is to define the objectives of the system. These objectives should be traceable to the customers of the system. To ensure executability was identified as the primary objective. The matrix is similar to an objective tree starting from the upper right and cascading down and left. For systems dynamicists, the objectives could serve as sectors, since multiple activities are performed to support each objective.

The **X** in each box denotes that there is a relationship between two variables. Matrix 2 is an intra-relational matrix, where the **X** indicates a relationship between objectives.

Defining the objectives that satisfy the customers is essential for all quality management organizations. This exercise enabled management to focus on the high value areas. Gathering information to populate these matrices accurately often requires communication and feedback from the customers.

Matrix 3 - Identify Activities

Each activity performed by an organization should be traceable to an objective. Activities that do not directly support the objective should be eliminated. Matrix 3 represents the activities that an organization performs to support each objective. Group brainstorming and other knowledge elicitation methods can be used to capture the tasks that must be performed.

Matrix 3 illustrates the traceability between the objectives and the activities performed by the organization. The **X** indicates a relationship.

IPT/SPO Activities \ OBJECTIVES	Max.Mnpwer Effctvne	Gather Information	Monitor Performance	Inform Customer	Obtain Funding	Negotiate Contract	Manage Risk	Ensure Executability
Mitigate Cost Risk							X	
Mitigate Tech Risk							X	
Mitigate Schedule Risk							X	
Technical Problem Solving							X	
Negotiate Reqt's w/ User							X	
Process ECPs						X		
Communicate with SPO				X				
Answer Pentagon Inquiries				X				
Answer User Inquiries				X				
Answer GAO Inquiries				X				
Support Formal reporting				X				
Support Award Fee				X				
Perform CPAR				X				
Contract/EVM Fact Finding		X						
Scheduled Communication with Us		X						
Scheduled communication with Co		X						
Assess Cost Risk		X						
Assess Tech Risk		X						
Assess Schedule Risk		X						
Ratings	X							
Training	X							
Recognition	X							
Process Improvement	X							
Lean Initiatives	X							

Figure 2.8
Matrix 3: Objectives vs. Activities

Matrix 4 – Define the Agents Who Execute the Activities

Each activity requires some organizational function to execute the task. Some activities requires more than one (i.e. Identify Risks often requires engineering, management, and financial officers). Matrix 4 illustrates the agents required to perform each activity.

Multiple functional organizations are represented within the SPO organization. Agents represent the different functional areas and are defined as follows:

PM:	Program Management
EN:	Engineering
FM:	Financial Management
PK:	Contracting
ADMIN:	Administrative Support
DCMA:	Defense Contracting Management Agency

The EN column highlights the activities that engineers perform within the organization. An **X** defines the activities each agent is responsible to perform in order to meet the objective requirements.

IPT/SPO Activities	PM	EN	FM	PK	ADMIN	DCMA
Mitigate Cost Risk		X	X			
Mitigate Tech Risk		X				
Mitigate Schedule Risk		X	X			
Technical Problem Solving		X				
Negotiate Reqt's w/ User	X	X				
Process ECPs	X	X		X		
Communicate with SPO	X	X	X	X		
Answer Pentagon Inquiries	X	X	X	X		
Answer User Inquiries	X	X	X	X		
Answer GAO Inquiries	X	X	X	X		
Support Formal reporting	X	X	X	X		
Support Award Fee	X	X	X	X		
Perform CPAR	X	X	X	X		
Contract/EVM Fact Finding	X	X	X	X		
Scheduled Communication with User	X					
Scheduled communication with KTR	X	X	X	X		
Assess Cost Risk		X	X			
Assess Tech Risk		X				
Assess Schedule Risk		X	X			
Ratings	X					
Training	X					
Recognition	X					
Process Improvement	X					
Lean Initiatives	X	X	X	X	X	X

Figure 2.9
Matrix 4: Activities vs. Agents

Matrix 5 – Identify System Workload Drivers (Customer Interest/Involvement)

Every management system provides products and services for customers. Customer interests and involvement drives the workload with positive and negative information feedback. Matrix 5 defines the associations between activity and customer.

Matrix 5 highlights several of the workload drivers for engineering workload in a SPO. An **X** is used to indicate a relationship between the customers and the workload drivers associated with the customers' interest and involvement with the system.

CUSTOMERS	Customer Interests/Involvement		GAO Inquiries	Pentagon Inquiries	User Inquiries	Political Satisfaction	Pentagon Satifaction	User Satisfaction	GAO Satisfaction	KTR Motivation	Confidence in SPO Leadership	Requirements Stability	Number of ECPs	Award Fee Percentage	KTR EVM Performance	Cost Risk Performance	Tech Risk Performance	Schedule Risk Performance	Contract Complexity	Technical Maturity	Funding Stability	National Exposure	# of User Driven Requirement Changes	Confidence in Risk Assessment	Risk Reporting Accuracy	SPO/IPT Task Efficiency	SPO/IPT Producivity	Information Flow Efficiency	Job Satisfaction
SPO										X	X	X	X	X	X	X	X	X	X	X		X	X	X	X	X	X	X	X
KTR									X		X		X	X	X	X	X	X		X		X	X	X			X		
DC			X		X	X		X		X			X	X	X	X	X	X		X	X		X	X			X		
USER				X			X			X		X	X	X	X	X	X		X	X			X	X			X		
GAO																													

Figure 2.10
Matrix 5: Customers vs. Customer Interests/Involvement

Matrix 6 – Identify the Dynamic Relationship between Workload Drivers and Activities

Matrix 6 represents what I call a causal inter-relational matrix (Cintergram). The word causal refers to the dynamic relationship that exists between the variables. The **X** is replaced with a "+" or "-", which represents reinforcing or balancing relationship between variables.

The information illustrated in Matrix 6 reveals the causal relationships between the activities and the workload drivers. For example, Pentagon Inquires relates to the activity

Answer Inquiries. Thus, a greater number of Pentagon Inquiries necessitates an increase in the manpower required to Answer Inquiries. A "+" is used to identify this relationship

Depending on the number of interdependencies found within the system, only a small percentage of the squares should be filled. It is not uncommon for some variables to a have a very weak correlation.

IPT/SPO Activities	Customer Interests/Involvement	GAO Inquiries	Pentagon Inquiries	User Inquiries	Political Satisfaction	Pentagon Satifaction	User Satisfaction	GAO Satisfaction	KTR Motivation	Confidence in SPO Leadership	Requirements Stability	Number of ECPs	Award Fee Percentage	KTR EVM Performance	Cost Risk Performance	Tech Risk Performance	Schedule Risk Performance	Contract Complexity	Technical Maturity	Funding Stability	National Exposure	# of User Driven Requirement Changes	Confidence in Risk Assessment	Risk Reporting Accuracy	SPO/IPT Task Efficiency	SPO/IPT Producivity	Information Flow Efficiency	Job Satisfaction	Age of Workforce	Producitivity	Turn Over Rate
Mitigate Cost Risk															-	-	-														
Mitigate Tech Risk															-	-	-														
Mitigate Schedule Risk															-	-	-														
Technical Problem Solving (TIMs)																															
Negotiate Requirements w/ User																+															
Process ECPs												+															+				
Communicate with SPO		+	+																								+				
Answer Pentagon Inquiries (unscheduled work)			+																								+				
Answer User Inquiries (unscheduled work)																															
Answer GAO Inquiries (unscheduled work)		+																													
Support Formal reporting (Briefings, MAR, DAES)																															
Support Award Fee																															
Perform CPAR																															
Contract/EVM Fact Finding														-	-											+					
Scheduled Communication with User											-					-	-				+					+					
Scheduled communication with Contractor														-	-	-	-				+					+					
Assess Cost Risk															-	-	-									+					
Assess Tech Risk															-	-	-									+					
Assess Schedule Risk															-	-	-									+					
Ratings																											+				
Training																											+				+
Recognition																											+				
Process Improvement																											+		+		
Lean Intiatives																													+		

Figure 2.11
Matrix 6: Activities vs. Customer Interests/Involvement

Matrix 7 – Identify the Measures of Performance for the Activities

Once the Activities are defined, the next step is to define the measures of performance for each activity. For the system modeler, identifying the measures of performance for each of the activities will provide good data to be used to validate and verify the simulation models. For management, defining the measures of performance for each activity provides essential insight to internal business performance. These metrics, if tracked, can enable the program manager to identify and focus on the areas of improvement within their organization.

Matrix 7 illustrates the associations between activities and corresponding activity performance measures. Each engineering activity was given a meaningful measure of performance. Little data was available for several of these activities. However, as a result of the activity, management was motivated to start tracking performance metrics to ensure they were satisfactorily executing the required activities.

Activity Performance Measures	Performance/Requirements Gap	% Complete of Risk Mitigation Plans	% Complete of Technical Risk Assessm	% Complete of Cost Risk Assessment	% Complete of Schedule Risk Assessm	% Complete of Management Feedback	% Complete of CPAR	% Participation in Award Fee	% Timely formal reporting	% of Timely ECP resolution	% of Timely DR resolution	% on time tasker Responses	% Tasker Rework	% of workweek spent communicating wi	% Complete for Recognition	% of Personnel Meeting Training Requir	% unplanned work	
IPT/SPO Activities																		
Assess/Mitigate Cost Risk		X																
Assess/Mitigate Tech Risk		X																
Assess/Mitigate Schedule Risk		X																
Technical Problem Solving																		
Negotiate Requirements w/ User	X																	
Process ECPs										X								
Staff Meetings																		
Answer Pentagon Inquiries												X	X					
Answer User Inquiries												X	X					
Answer GAO Inquiries												X	X					
Formal reporting (MAR, DAES)									X									
Support Award Fee								X										
Perform CPAR							X											
Contract/EVM Fact Finding																		
Scheduled Comm. with User																		
Scheduled comm. with KTR						X								X				
Assess/Mitigate Cost Risk				X														
Assess/Mitigate Tech Risk			X															
Assess/Mitigate Schedule Risk					X													
Ratings																	X	
Training																X		
Recognition															X			
Process Improvement																		X
Lean Initiatives																		

Figure 2.12
Matrix 7: Activities vs. Activity Performance Measures

Matrix 8 – Identify the Dynamic Relationships between Activity Performance Measures and Workload Drivers (Customer Interests and Involvement)

Matrix 8 is another causal intra-relational matrix. It identifies the causal relationships between the activity measures of performance and the workload drivers. For example, if the management organization has a high *% of on Time Tasker Responses* (User Inquires), then there is a positive feedback to User Satisfaction denoted by the "+" in the matrix box. Identifying the causal relationships for activity measures and the workload drivers can be done in a brainstorming session with system experts or with customers in a customer feedback session.

Determining the nature of these relationships requires more in-depth study, data collection, and expert knowledge elicitation.

Activity Performance Measures Customent Interest/Involvement	Performance/Requirements Gap	% Complete of Risk Mitigation Plans	% Complete of Technical Risk Assessm	% Complete of Cost Risk Assessment	% Complete of Schedule Risk Assessm	% Complete of Management Feedback	% Complete of CPAR	% Participation in Award Fee	% Timely formal reporting	% of Timely ECP resolution	% of Timely DR resolution	% on time tasker Responses	% Tasker Rework	% of workweek spent communicating w	% Complete for Recognition	% of Personnel Meeting Training Requir	% unplanned work
GAO Inquiries																	
Pentagon Inquiries																	
User Inquiries																	
Political Satisfaction	+								+			+					
Pentagon Satifaction	+								+			+					
User Satisfaction	+									+		+					
GAO Satisfaction										+	+	+					
KTR Motivation						+	+	+						-			
Confidence in SPO Leadership																	
Requirements Stability																	
Number of ECPs																	
Award Fee Percentage																	
KTR EVM Performance																	
Cost Risk Performance																	
Tech Risk Performance																	
Schedule Risk Performance																	
Contract Complexity																	
Technical Maturity																	
Funding Stability																	
National Exposure																	
# of User Driven Requirement Changes																	
Confidence in Risk Assessment			+	+	+	+											
Risk Reporting Accuracy																	
SPO/IPT Task Efficiency																	
SPO/IPT Producivity																	
Information Flow Efficiency																	
Job Satisfaction																	-
Age of Workforce																	
Producitivity																	
Turn Over Rate																	
Competency																+	
Morale															+	+	-

Figure 2.13

Matrix 8: Activity Performance Measures vs. Customer Interests/Involvement

Matrix 9 – Identifying the Workload Drivers (Customer Involvement/Interests) Causal Inter-relational Diagram:

Matrix 9 is also a causal inter-relational matrix (Cintergram). This matrix identifies the causal relationships intrinsic to the workload drivers that will affect the activities performed by the management system. For example, User Satisfaction is a workload driver that affects the number of program inquiries the user requires of a SPO. Greater user satisfaction yields a lower number of inquires. Therefore, a "-" is placed in the matrix box intersecting User Inquiries and User Satisfaction.

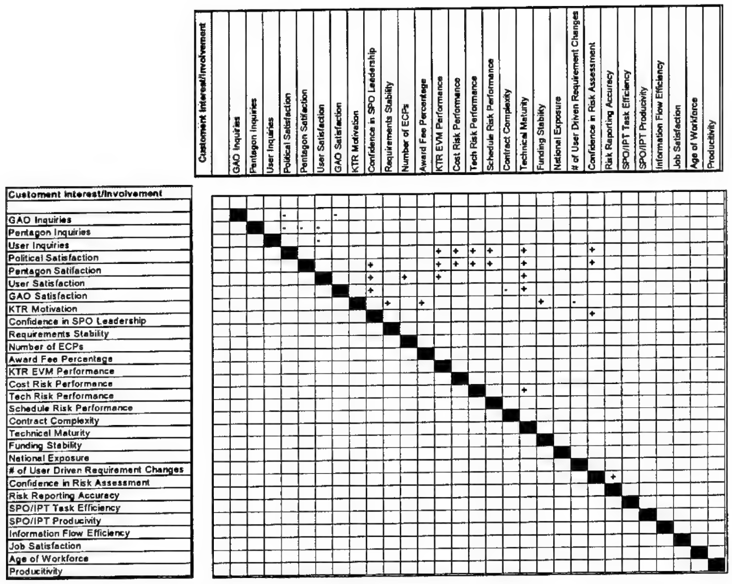


Figure 2.14
Matrix 9: Customer Interests/Involvement Cintragram

Boundary Definition Using MCM:

Once the original MCM was completed, it was obvious that many of the variables could be pooled together and that others were relatively insignificant to the engineering workload. Based on further interviews, the MCM was simplified to the highest meaningful level of aggregation. Below are the revised matrices.

<u>Revised MCM</u>

After completing the MCM, the two objectives the engineering workforce primarily supported were to administer information and manage risk. Engineers supported other objectives, but an interview revealed that over 90 percent of the effort engineers expend was in these two areas.

Previously, the team identified 24 activities that engineers performed. These activities were simplified into 3 areas: identify risk, mitigate risk, and administer information.

Matrices 4, 5, and 6 are also simplified to address the high yield areas for engineering manning. The workload drivers, or Customer Interest/Involvement, were reduced from 33 to 13. Figures 3 and 4 show the remaining revised matrices used for generating the initial stock and flow structure.

Figure 2.15 illustrates the revised matrices 4, 5, and 6.

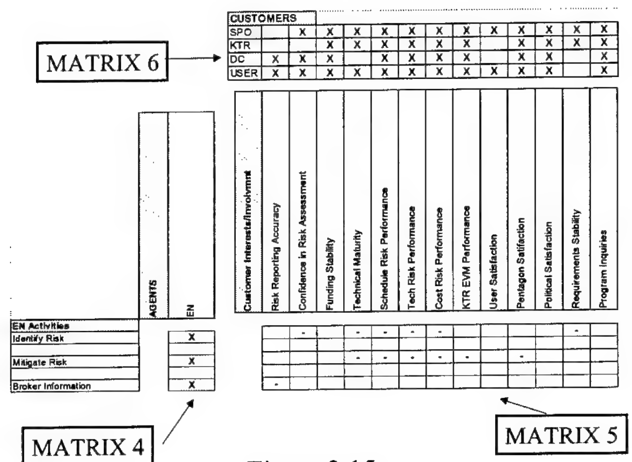


Figure 2.15
Matrices 4, 5, and 6 Revised

Figure 2.16 illustrates matrices 7 and 8.

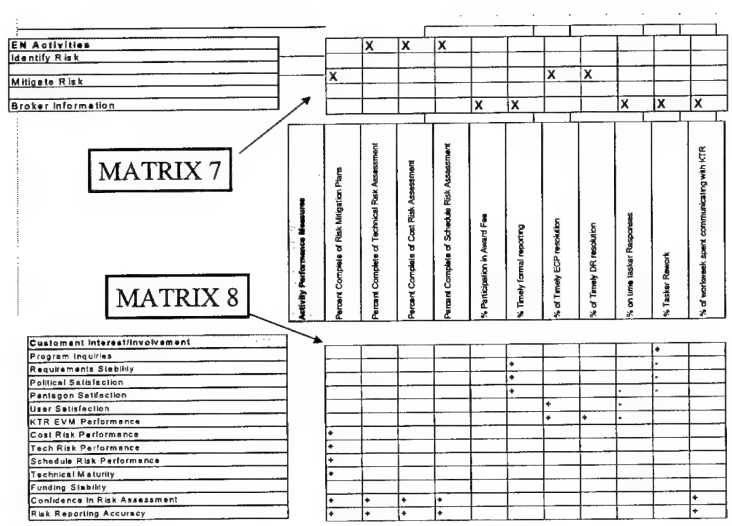


Figure 2.16. Matrices 7 and 8 Revised

Figure 2.17 illustrates the revised matrix 9.

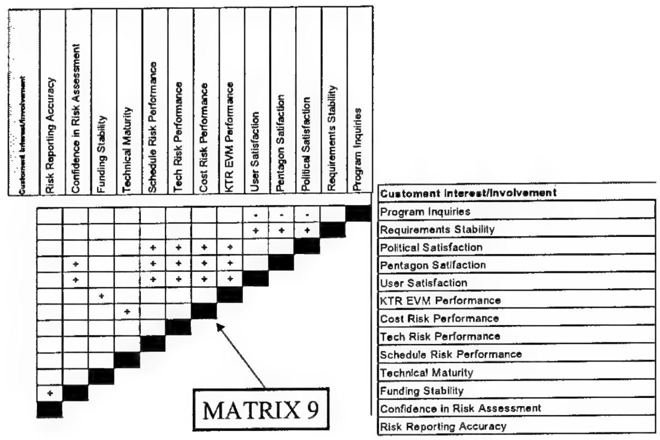


Figure 2.17. Revised Matrix 9

MCM Example:

In this section, I will present an example that walks through the entire MCM. I elected to trace the dynamic behavior for Identify Risk. This activity supports the objective Manage Risk highlighted by Matrix 1 in Figure 2.18. The function, Manage Risk, is an engineering function valued by all of the system customers. This is represented in the highlighted section of Matrix 1.

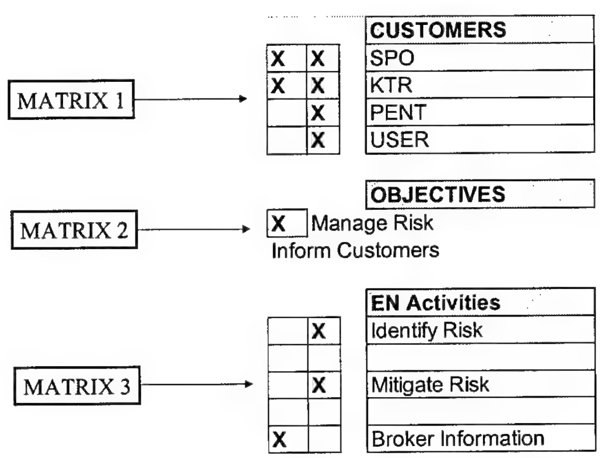


Figure 2.18 Matrices 1, 2, and 3 Revised

In figure 2.19, Matrix 4 highlights the causal relationships that interact with the **Identify Risk** Activities. Matrix 5 reveals the causal relation identified between **Technical Maturity** and Identify Risk. A higher the level of **Technical Maturity** requires less engineering effort to **Identify Risk.**

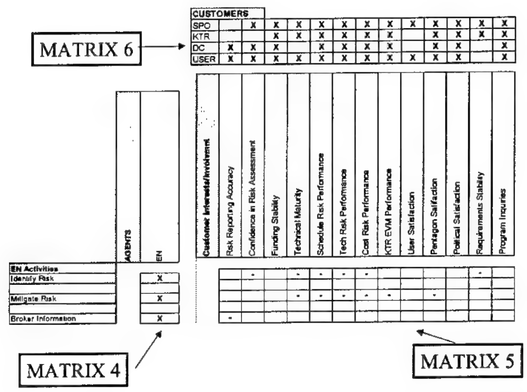


Figure 2.19 Matrices 4, 5, and 6 Revised

In figure 2.20, matrices 7 and 8 show several relationships. **% complete of Technical Risk Assessment** is function of number of engineers to **Identify Risk**. If the proper number of engineering staff is allocated to **Identify Risk** then the **% complete of Technical Risk Assessment** performance is good. If this activity performance measure is high, then the variable, **Confidence in Risk Assessment**, is also high. This is represented by the "+" symbol.

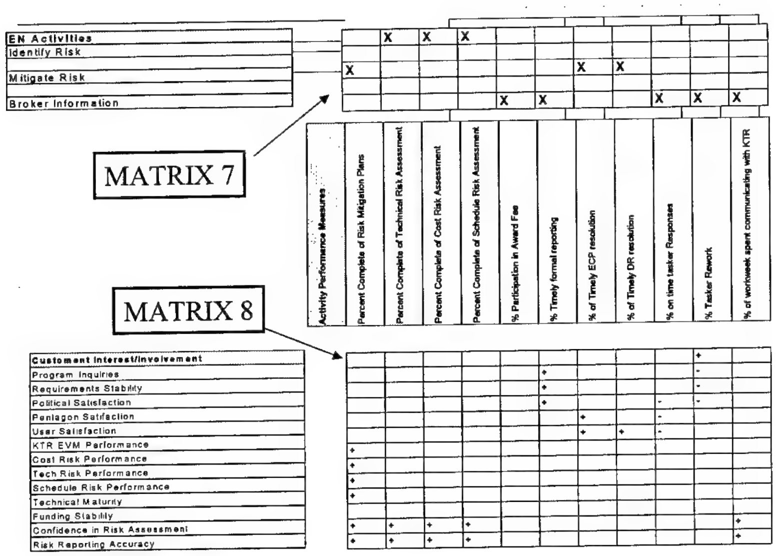


Figure 2.20. Matrices 7 and 8 Revised

In the revised matrix 9, the previously complex Cintragram is greatly simplified by raising the level aggregation. Figure 21 highlights the relationship between **Confidence in Risk Assessment** and **Risk Reporting Accuracy**. The “+” indicates that the greater the **Confidence in the Risk Assessment** the higher the **Risk Reporting Accuracy**. We can follow the matrix to show that the feedback to the **Customer’s Satisfaction** is positive the higher **Risk Reporting Accuracy**. This in turn affects the **Number of Inquiries** engineers will be required to answer.

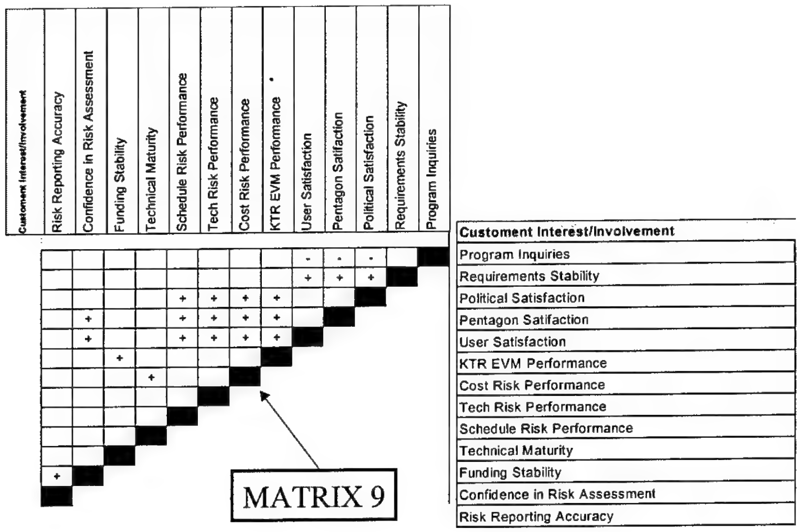


Figure 2.21 Matrix 9 Revised

<u>Stocks and Flows Example:</u>

This section describes a stock and flow diagram used to model a portion of the information generated by the MCM, which was described in the section above. **Technical Maturity** is a variable that determines the **Required Effort to Identify Risk**. The more mature the program, the less engineering manpower is required to identify risk. **Confidence in Risk Assessment** compares the **Required Effort to Identify Risk** with **Available Manpower to Identify Risk**. If the **Available Manpower to Identify Risk** meets the defined threshold, the engineers will achieve an acceptable **Rate of Risk Discovery**. **Rate of Risk Discovery** determines the amount of **Discovered Risk** for each time interval. If the **Discovered Risk** matches the **Discovery Profile** it increases the **Risk Reporting Accuracy**.

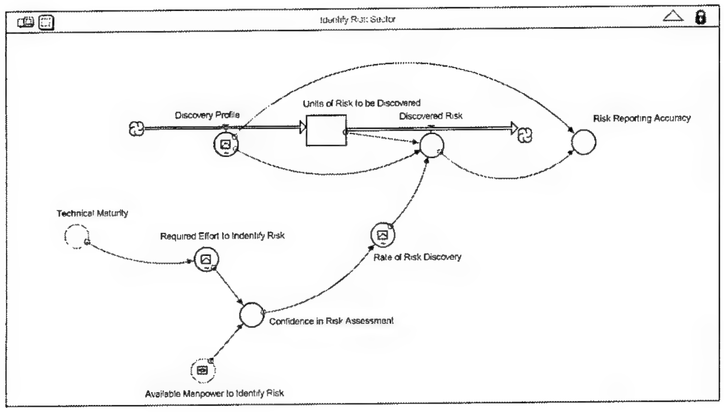


Figure 2.22. Identify Risk Sector

The sector displayed above is one portion of a larger stock and flow model of the entire the engineering manpower system defined by the MCM.

<u>MCM Summary</u>

MCM provides a rigorous process of assembling information during the information-gathering phase of model development. In addition, the organized structure gives users a greater insight into the many interrelationships and interdependencies found within an extremely complex system. The process of defining a MCM cultivates a strong, intuitive appreciation for the entire system. Through better understanding of the entire system, a modeler can simplify the process boundary definition and the degree of aggregation. The next task was to explore the dynamic relationships and information feedback found with in the system.

Individual Interviews:

Several interviews were performed with experts in the government engineering community. Interviewees included the following individuals: the USAF Aeronautical Systems Engineering Director and former F-22 Program Chief Engineer, the USAF Aeronautical Systems Chief

System Engineer, the former F-22 SPO Chief Engineer, the Director of Engineering of the USAF Aging Aircraft Program, and the current F-22 SPO Deputy Director.

Prior to the interview process, several key dynamic relationships were isolated to present to the interviewees. Each interviewee was permitted to describe the behavior of the presented interrelationship and provide further comments in other areas that they felt needed to be addressed. The interviewing process was iterative, and each interviewee was allowed to see the comments of the other individuals. As a result, the individuals reached a consensus on the critical behaviors to be modeled. The method was similar to Ford and Sterman's expert knowledge elicitation method (Ford et al:1998).

Define System Behaviors:

Behaviors identified during the interview sessions are identified in the following sections.

User Requirements Over Time:

This variable looked at the change in user requirements over time. Many weapon system development activities begin with a fuzzy set of user requirements proposed for the development. For aircraft development, many requirements include technologies that are unproven and extremely high risk. As a result, program managers and governmental agencies often manipulate the requirements during the development phase.

Figure 2.23 is a graphical representation of varying user requirements. Above 0 indicates a net gain of requirements or a requirement increase. Below 0 indicates a decrease.

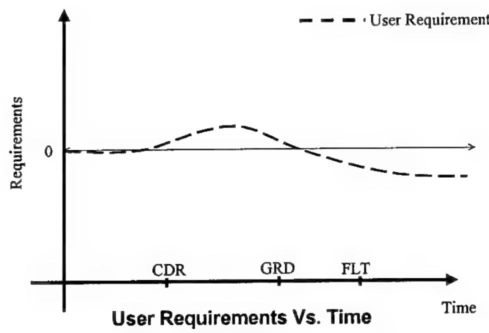


Figure 2.23 User Requirements Behavior

CDR represents the critical design review. GRD, represents ground testing, and FLT represents the beginning of flight test. The graphic shows that user requirements are generally stable until CDR. CDR is the first time that the program clearly communicates to the user the design proposed to meet the user requirements. Many times, there are discrepancies in requirement definition, and requirements are often increased. The CDR timeframe is also one of the last times in the process that the user can add work to the contract before it is too costly. This is why a slight increase in requirements is illustrated.

Ground testing, GRD, is a significant milestone for many programs in that it signifies that the first articles have been produced. Immature technologies and gold plating requirements are addressed. The users often relieve requirements to an acceptable level of risk, due to cost and schedule constraints. This is illustrated by the slight decrease of the overall system requirements during the last half of the development program.

Technical Maturity vs. Time:

Technical maturity, for most programs, has a goal-seeking behavior as seen in Graph 2.

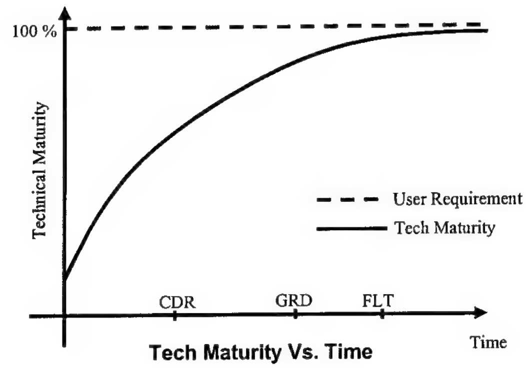


Figure 2.24 Tech Maturity Behavior over Time

This graph describes the percent of the system meeting the user requirement. This is a highly aggregate representation of the system. The inherent risks of multiple subsystems interacting makes this metric almost impossible to quantify since many portions of the system exceed user requirements and others will never approach the desired capability. As a whole, the interviewees accepted this to be an accurate representation of technical maturity over time.

Program Risk vs. Time:

Risk performance versus time is another dynamic relationship that is difficult for which to gather data and quantify. Program risk is defined as the unidentified, "unknown, unknown" factors inherent to the system, which affect cost, schedule, and performance. The behavior of this variable is entirely system dependent. The role of government engineers is to provide expertise to identify these risks and to help the contractor mitigate them.

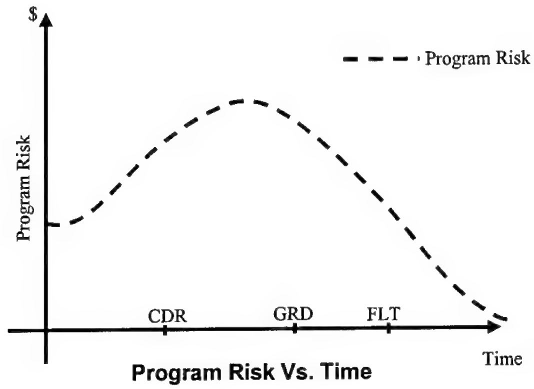


Figure 2.25 Program Risk Behavior over Time

Figure 2.25 illustrates the behavior over time. Initially, a development effort has identified a certain amount of programmatic and technical issues that are labeled as risk. Depending on the maturity of the technologies associated within the system, risk is identified through CDR until GRD. Once the initial development articles are produced, fewer risks are identified. The development team, in concert with the users, will look at requirement relief and the application of management reserve to mitigate the risk. The result is marginal risk as the development program begins to transition into production.

Number of Program Inquiries vs. Time:

For highly visible and politically charged military programs, information brokerage is one of the most important services a SPO provides. Engineers who understand the critical details of system performance and technical risks are called upon to answer other government agencies and Congress. The interviews reveal that answering inquires produces an S-curve behavior. In the beginning of a program, there is little external interference due to the immaturity of the weapon system. CDR is when programs generally start seeing inquiries. These inquiries come from various groups and generally deal with design critiques, new requirements, and new cost, schedule, and performance estimates. Programs offices see an exponential growth and then a

tapering of inquiries as the program approaches Milestone III and the production contract decision, the biggest decision for any acquisition effort.

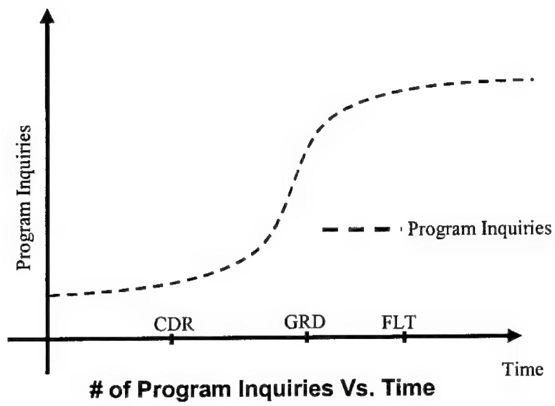


Figure 2.26 Program Inquiry Behavior over Time

Engineering Manpower vs. Time:

The next relationship presented in the initial interview was the allocation of engineering manpower over time.

Figure 2.27 illustrates the common approach to engineering manpower that reflects the contractor's manpower. Initially, manpower is added to the program at the beginning of the program and reaches a peak near the CDR. Once the design is set both the government and contractors dissolve the engineering workforce by two/thirds at completion.

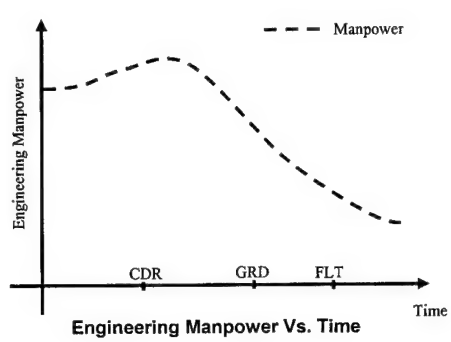


Figure 2.27 Engineering Manpower Burndown vs. Time

Cost to Mitigate Risk vs. Time:

This relationship illustrates the cost associated with making changes and the mitigation of risks during the development phase. Until CDR, the cost of mitigating risks is relatively low. After CDR, the same risks become increasingly more expensive. Early identification and mitigation of risk is essential in minimizing program costs.

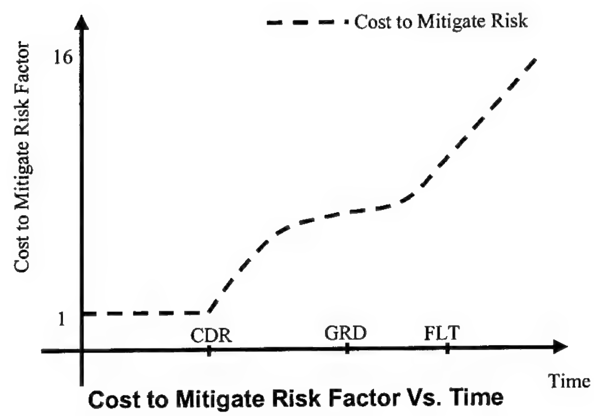


Figure 2.28 Cost to Mitigate Program Risk over Time

Risk Impact to Schedule vs. Time:

Similar to the cost factor previously mentioned, there is a varying impact of risk to a program's schedule. The earlier risk is identified, the less the impact. Figure 2.29 was generated by the interviews. The curve is identical to the Cost to Mitigate Risk Factor mentioned above.

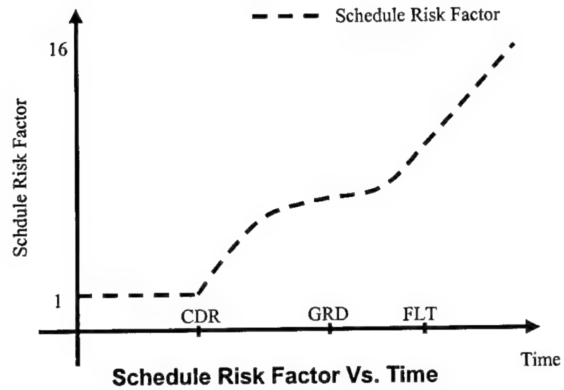


Figure 2.29 Schedule Risk Factor over Time

Methodology Summary:

This section describes the methodology required to generate the stock and flow structure used to model the system. F.A.S.T. was used as an efficient method for group knowledge elicitation. The information generated by the F.A.S.T. was organized using the MCM. In addition to providing structure to the system, the MCM proved to be an excellent tool for determining the scope and boundaries of the system. Once the system was understood, experts were asked to describe the behaviors of the system.

The next chapter will fully describe the stock and flow structure generated to capture the system. Appendix A is provided a quick overview of the different elements of the system dynamic structure. Appendix D provides the mathematical equations associated with the structure.

3. Model Description

Once the information-gathering phase was completed, the I-think® computerized system dynamic modeling tool was used to develop a stock and flow structure. The development of the simulation model was an iterative process. An object-oriented approach was used in which individual sectors were developed and validated and then incrementally integrated with other sectors. There were nearly 30 iterations and several expert interviews performed during the I-think development process. The following section describes the system dynamics sectors that were modeled.

Requirements Sector:

This sector addresses the user requirements over the life of the development program. The **New Requirements** variable is an input to the system and is an independent factor. Requirements are reduced when the **User Willingness to Relieve Requirements** threshold is exceeded. The **Business Performance** input (earned value) is the variable that affects the users willingness to relieve the requirements. If the Earned Value performance is bad, then the **User Willingness to Reduce Requirements** increases. The **Performance Requirement Gap** represents the difference between **Tech Maturity** and the user **Requirements** goal.

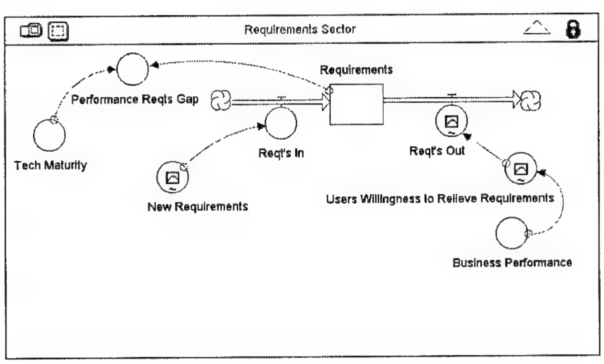


Figure 3.1: Requirements Sector

Identify Risk Sector:

This sector models the identification of risk within a government program. The premise is this: If the engineering support is manned with the appropriate number of individuals, the SPO will identify risk at an acceptable rate. If there are too few individuals, then risk identification is slow. Based on the behaviors presented earlier, the earlier risk is identified, the smaller the impact to program cost. The longer it takes to identify a risk, the more expensive it is, by a factor of ten, to mitigate.

Discovery Profile is an input variable, which is a time-phased input to **Units of Risk to be Identified**. The **Performance Reqts Gap** input drives manpower planning. Most SPOs determine the number of individuals required on a program by the maturity of the system. For a more mature system, fewer engineers are required to identify risks. **Available Manpower to Identify Risk** is the actual engineering manpower allocated by the SPO. The delta between actual and required yields the **Confidence in Risk Assessment** variable. This factor determines the **Rate of Risk Identification** variable that is the output factor for the **Units of Risk to be Identified.**

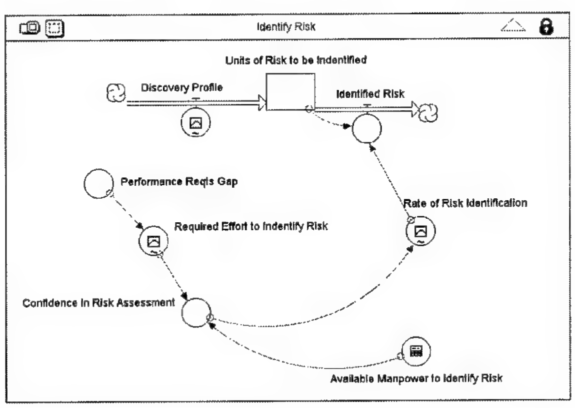


Figure 3.2 Identify Risk Sector

Mitigate Risk Sector:

This is the most complex sector in the program. The sector describes the flow of risk in a program. There are two ways for programs to address risk. First, management reserve can be applied to risk, thereby eliminating cost increase. Second, the SPO and contractor can work to mitigate risk or cost growth potential through "other means," including technical problem solving, business practices, etc.

There are two major stocks in this sector, **Reported Program Risk** and **Management Challenge**. **Reported Program Risk** is the amount of risk that is reported external to the program. **Management Challenge** represents risks that currently exist yet are under mitigation.

The inputs to the **Reported Program Risk** stock come from the different inputs of risk into the system. These inputs of risk include the **Identified Risk** variable from the Identify Risk sector; **Risk Increase Due to New Requirements,** which represents additional costs due to the user increasing the weapons system requirements; **Unmitigated Risk,** which represents the risk that was unable to be mitigated; and **Mitigated Risk Out,** which represents one-third of the mitigated risk that reenters the system. The outflow is the **Management Reserve** that is applied to the risks. Another pathway for **Program Risk Out** is the **Requirements Relief Risk Out**. If the user relieves requirements, then risk is reduced from the program at a rate defined as **Risk Relief Due to Reduced Requirements**.

The **Management Challenge** input is dependent on the **Selected Management Challenge** factor. This is a time-phased factor in which, early in a program, the SPO accepts a larger percentage of risk as management challenge. This is because there is time to mitigate the risks. As the program matures and the design is well established, less management challenge is applied due to the decreased likelihood to reduce the program costs. There are two outputs to the

Management Challenge stock: the **Unmitigatable Risk** and **Mitigate Risk Out.** The **Unmitigatable Risk Factor** is a time-phased variable that controls risk mitigation. Early in the program, there is a much higher probability to mitigate risk than later in the program. The **Mitigate Risk Out** variable is governed by manpower allocation.

The **Management Challenge** figure dictates the number of engineers required to mitigate risk, defined as **Manpower Required**. The **Available Manpower to Mitigate Risk** dictates the number of engineers allocated to mitigate risk. These two variables are computed to determine the value of **Confidence in Risk Mitigation**. This variable governs the **Mitigate Risk Factor,** which determines the **Mitigate Risk Out**.

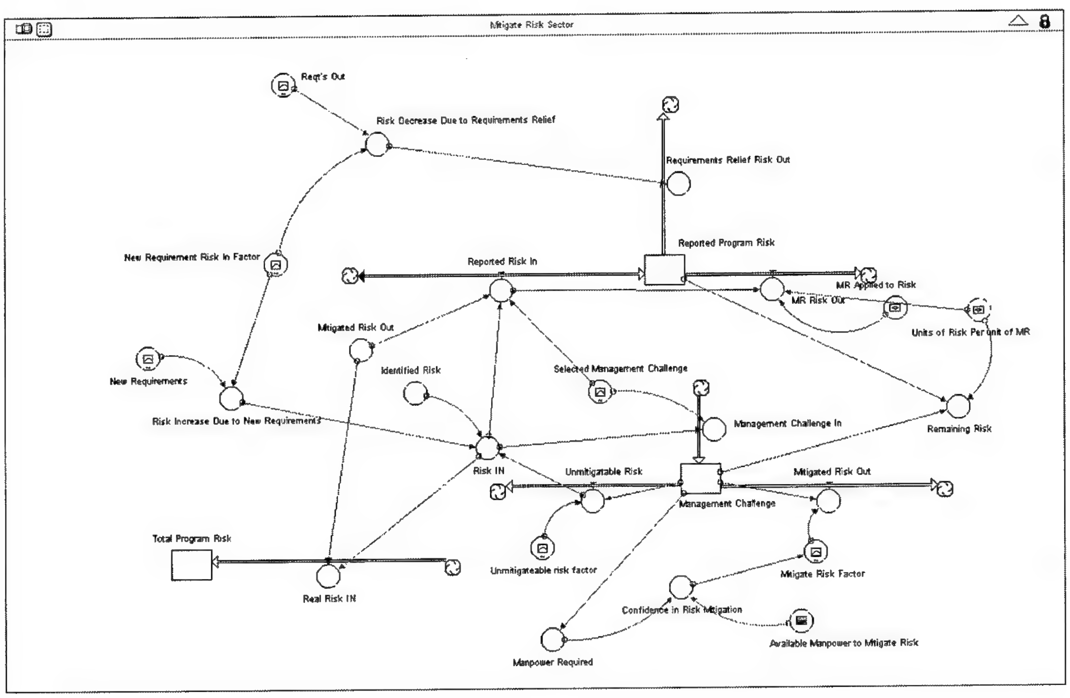


Figure 3.3 Mitigate Risk Sector

Other factors in the sector include **Remaining Risk,** which represents the sum of **Management Challenge** and **Program Risk**. **MR Applied to Risk** represents the number of units of management reserve to be applied to risk. **Units of Risk Per Units of MR** is a

conversion factor between risk and management reserve. In reality, both risk and management are measured in dollars. This simulation represents the value of one unit of risk to be substantially less than a unit of management reserve by a factor of six. **Total Program Risk** is a measure of the total risk associated with the program.

Technical Maturity Sector:

Tech Maturity is a goal-chasing curve and is an input into the system. **Initial Maturity** is the maturity already achieved prior to the beginning of the development program. The **Requirements** stock comes from the requirements sector, and is the variable that changes when new requirements are added and subtracted. **Program Schedule** represents the number of quarters the program office is planning to complete development. **Projected Capability** is the percentage of the requirements the program expects to deliver to the user. When a program begins development, many times meeting the full set of user requirements is an impossible task. The user is reluctant to reduce requirements and is willing to progress with the program to push the envelope of technology despite knowing that a 100% solution is unreasonable. The user is also very unlikely to reduce requirements due to the concern of losing more than an acceptable amount of capability.

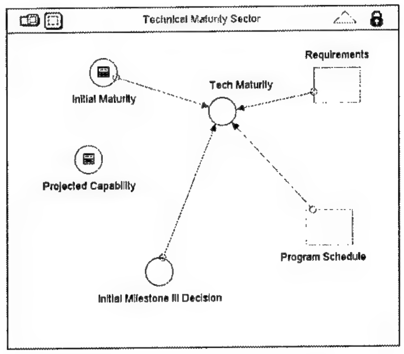


Figure 3.4 Technical Maturity Sector

Inquiries Sector:

This sector models the inquiries coming into a program office. There are three sources of inquiry inputs: **Planned Inquiries, Inquiries Due to Program Performance**, and **Inquiries Due to Customer Satisfaction**. **Planned Inquiries** are formal reporting inquires that each program office is required to perform. Reports such as the Monthly Acquisition Report (MAR), Defense Acquisition Executive Summary (DAES), and quarterly program reviews are examples of recurring reports that engineers support and are included in the **Planned Inquiries**. **Inquiries Due to Program Performance** are non-recurring inquires that are generated when the program is experiencing poor **Business Performance**. The **Inquiries Due to Customer Satisfaction** variable represents the additional inquiries that result from insufficiently supporting the inquiry workload. The Answer Inquiries Factor governs the workload and is calculated by the ratio of **Available Manpower to Answer Inquiries** and **Required Effort Inquires** variables. The **Required Effort Inquiries** is a predetermined value that is dependent on the number of inquires. The more inquires that are received, the more manpower that is required to answer inquiries. If the SPO does not allocate enough engineers to answer inquires, or **Inquiries Out**. **Inquiries**

Out represents the remaining factor that increases the number of **Inquiries Due to Customer Satisfaction** and is multiplied by the **Answer Inquiries Factor**.

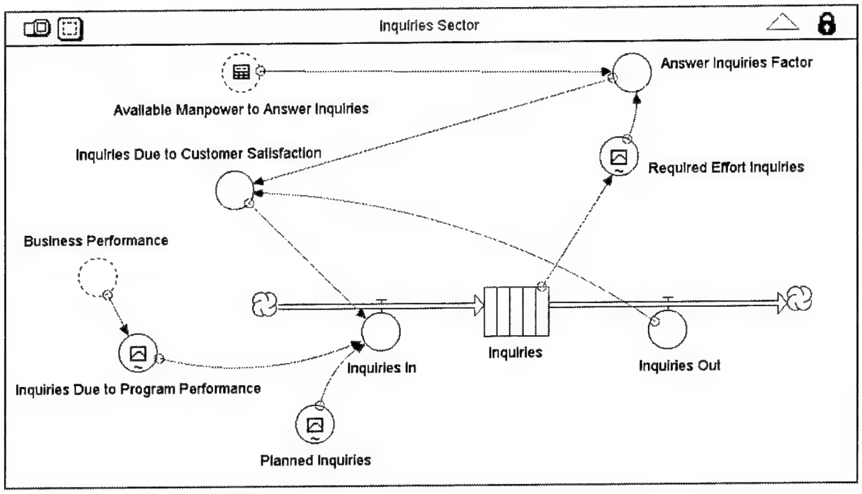


Figure 3.5 Inquiries Sector

Manpower Sector:

This sector contains the user-controlled input of manpower into the system. The **EN Manpower** variable is the sum of the allocated engineering manpower.

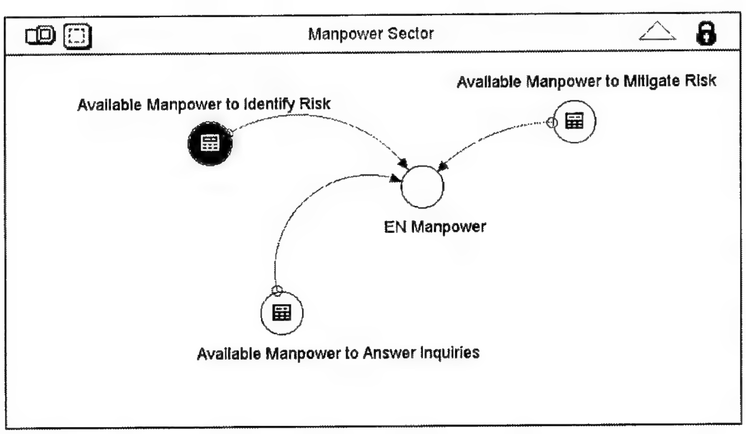


Figure 3.6 Manpower Sector

Cost/Schedule Performance Sector:

This sector models **Business Performance**, **Program Schedule**, and **Management Reserve**. **Business Performance** is the ratio of **budgeted at Completion (BAC)** and **Management Reserve (MR)** divided by **Estimate at Completion (EAC)** and **Reported Program Risk**, respectively. This ratio determines if the program is executable at the budget determined at the beginning of the program. If the value is less than 1, then the user must determine whether he is going to reduce requirements or seek additional funding. Other information feedback is determined by this variable. If the business performance is poor (<1) then the program could expect an increase in non-recurring taskers.

Program Schedule has an input and an output variable. **Schedule In** is a function of factors that increase a program's schedule. **Reported Risk In** is a variable that increases the schedule if risks are identified late in a program. Early identification of risk does not have a significant impact to schedule, whereas late risk identification is very costly and affects schedule. The other input to **Program Schedule** is the impact that new requirements have on schedule. The **Time Factor New Rqmt and Schedule 2,** in concert with the **New Requirement** input, yields an increase in schedule that is dependent on when a new requirement is added. The later a requirement is added to a program, the greater the impact it has on **Program Schedule**.

Mitigated Risk Out and requirement reduction variables govern **Schedule Out**. The **Mitigated Risk Out** compensates for the **Reported Risk In.** If the SPO allocates the proper effort to mitigate risk, some of the schedule impacts of the risks will decrease. There is also corresponding schedule relief associated with requirements; this is defined by the variables **Time Factor New Rqmt and Schedule** and **Reqt's Out**.

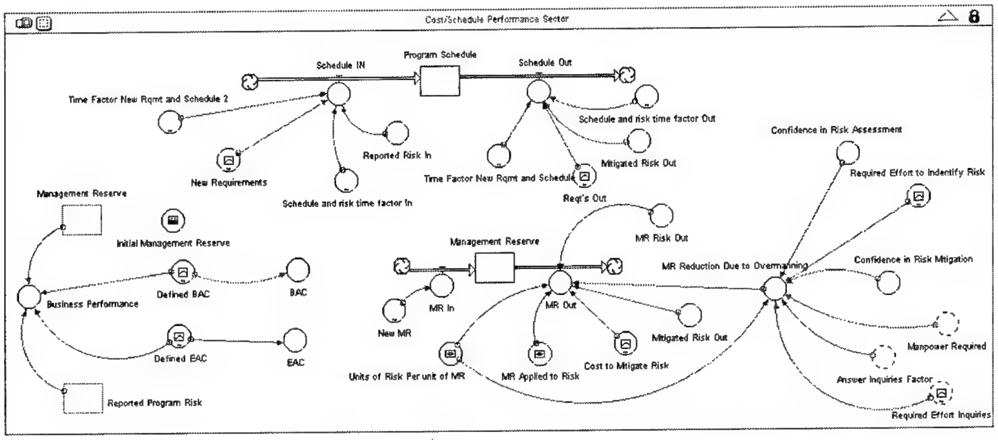


Figure 3.7 Cost and Schedule Performance Sector

The next key component of this sector is the **Management Reserve** region. Each program allocated approximately 10% of the contract cost for **Management Reserve**. Depending on the risk associated with the program, this variable can be greater or less than 10% and units can be added throughout the life of the program. The variable, **New MR**, is a controlled input.

MR Out represents the sum of the variables that drains the **Management Reserve** stock. These variables include; **MR Applied to Risk, MR Risk Out, Cost to Mitigate Risk,** and **Mitigated Risk Out**. **Cost to Mitigate Risk** is a time dependent variable and works in concert with **Mitigate Risk Out**. The two variables together establish that the point in program development at which risk is mitigated affects the cost of risk. The later in development a risk is mitigated, the greater the cost. The variable, **MR Applied to Risk,** is a control that allows the model user to reduce a significant amount of risk in a specific time increment. **MR Risk Out** is same output variable to the **Program Risk** stock. The variable **MR Reduction Due to Overmanning** penalizes the model user for overmanning the program. If the ratio of manpower required versus manpower allocated is exceeded, the program MR is reduced.

4. Model Analysis:

Due to the lack of substantial data, the model was not fit for performing critical statistical analysis. The main objective was to determine if the current structure and multiple assumptions yielded a reasonable response. Many scenarios were tested and the model was validated as a good representation of the system structure. I will examine one scenario with varying manning levels.

Adequate Engineering Support:

Below is several graph outputs from the model compared to the predicted behaviors with reasonable engineering manning levels.

As indicated in figures 4.1 and 4.2, the behavior of achieved technical maturity vs. time behaved very similarly to the predicted behaviors. Figures 4.3 and 4.4 illustrate the behaviors of program inquiries.

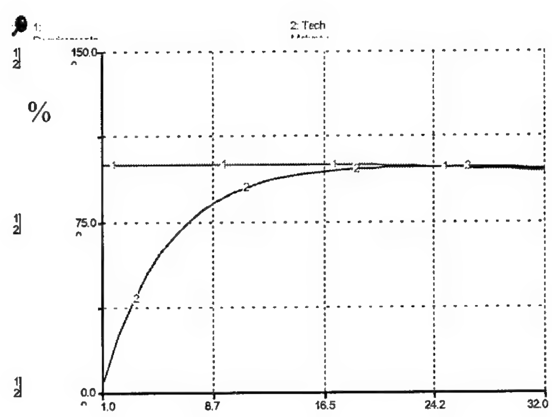


Figure 4.1
Simulated Technical Maturity vs Time

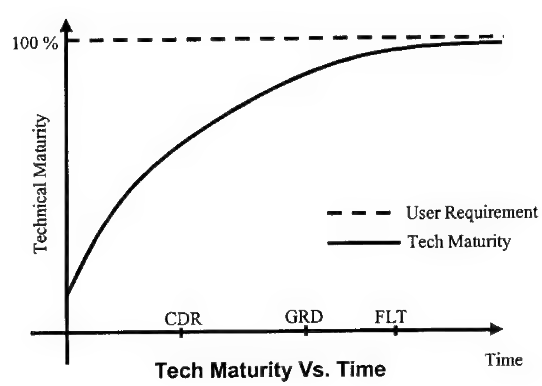
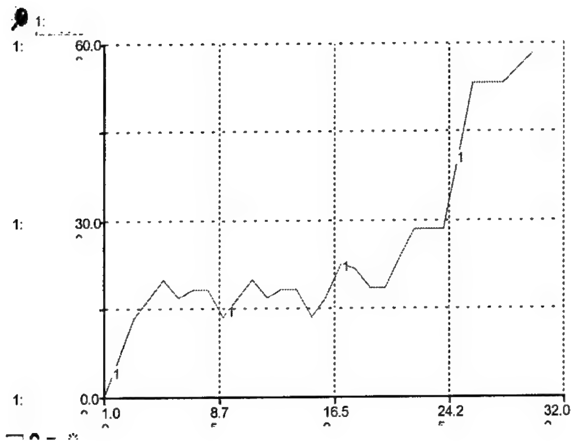


Figure 4.2
Predicted Technical Maturity vs. Time



Quarters

Figure 4.3
Simulated Inquiries vs. Time

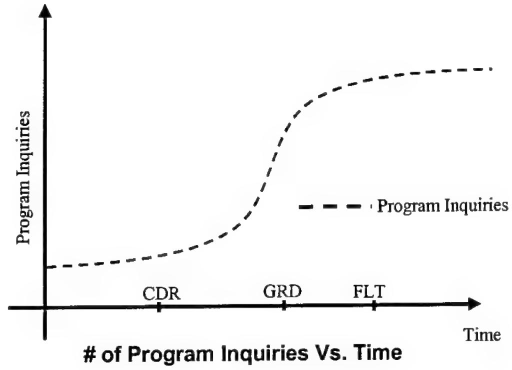
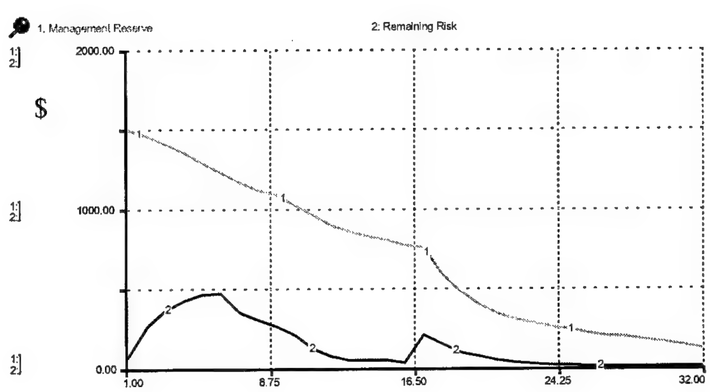


Figure 4.4
Predicted Inquiries vs. Time

Figure 4.5 shows **Management Reserve** versus **Remaining Risk** over time. The behavior represents a successful program where the difference between the **Management Reserve** and the **Remaining Risk** represents the amount of funds remaining during the development program. The two peaks on the **Remaining Risk** curves represent risk discovery in the system. According to the predetermined inputs, the engineering manpower identified risk in a timely fashion.



Quarters

Figure 4.5 Management Reserve and Program Risk over Time

Too Few Engineers:

The next graph represents a program office that has no engineers supporting the development activity. The program office is relying on the contractor to identify and mitigate risk. With all the variables remaining unchanged from the previous example, with the exception of reducing manpower, the results are seen in Figure 4.6.

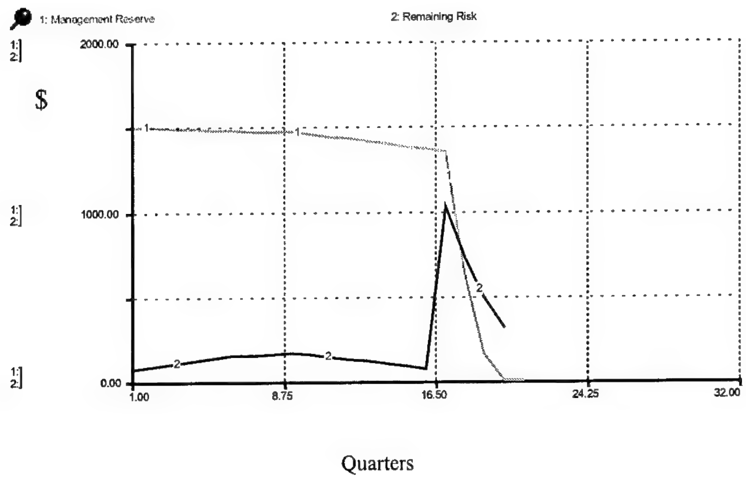


Figure 4.6 Management Reserve and Program Risk Performance with No Engineers

The results are as expected. The **Management Reserve** burndown is much slower, due to the very slow identification of risks represented by the **Remaining Risk** curve. The sudden peak a time=16, represents the contractor's identification of risks. The model is parameterized so that at the halfway point the contractor begins identifying risk. The program goes unexecutable at approximately time (T) =17, when the management reserve fails to cover the program risk.

Too Many Engineers:

The next graph represents the behaviors of Management Reserve and risk in a program office with too many engineers. The behavior is the opposite of the previous example. **Management Reserve** experiences a steep decrease due to the early identification and mitigation

of risks, and the extra costs of extra manpower. At T=16 there is only very slight increase to the **Remaining Risk** due to fact that contractors has very little risk to identify. Interestingly, the program becomes unexecutable at T-17 because the risk, though small, exceeds the remaining management reserve. The system model behaved as predicted by demonstrating the adversarial effects of overmanning.

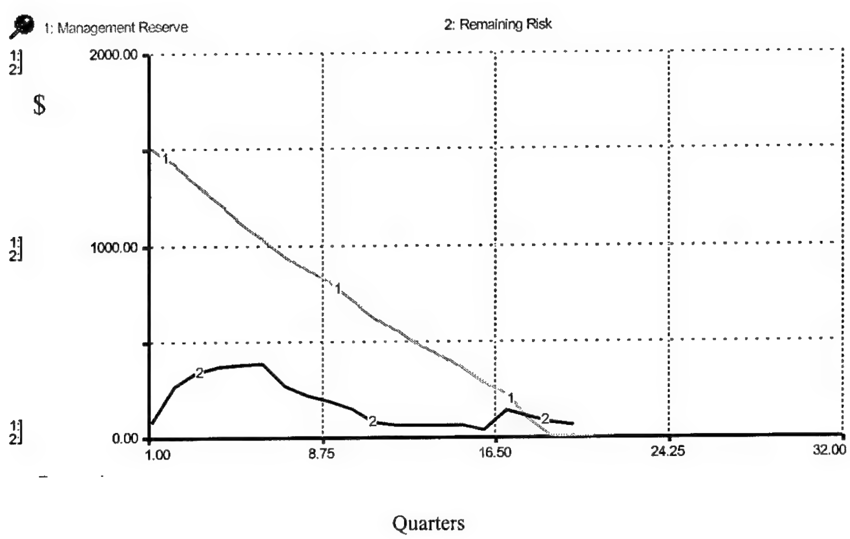


Figure 4.7
Management Reserve and Program Risk Performance with Too Many Engineers

Early Requirement Changes

The next scenario examined the effects of requirement changes early in the program. The parameters were reset to the adequate manning level described above. For the scenario, a 50% increase of requirements was added to the program in the first 8 quarters of the development cycle. Figure 4.8 illustrates the system behavior. The three step-like jumps at T=2, T=4, and T=6 represent increases in the requirement. The early changes to the requirements has little effect on the **Technical Maturity** which matures before the end of the program.

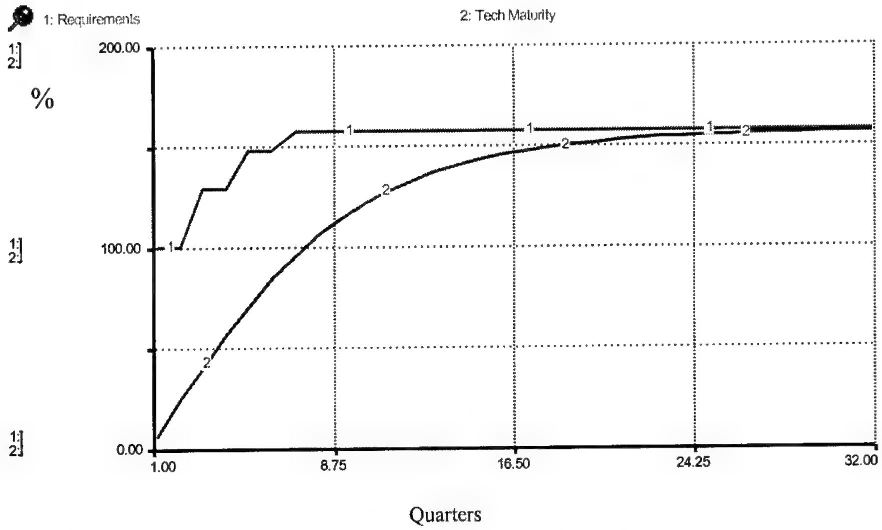


Figure 4.8 Tech Maturity and Requirement Behaviors with Early Increase in Requirements

Figure 4.9 illustrates the **Management Reserve** and **Risk** behaviors. Even with a 50% increase in the requirements, the program is executable.

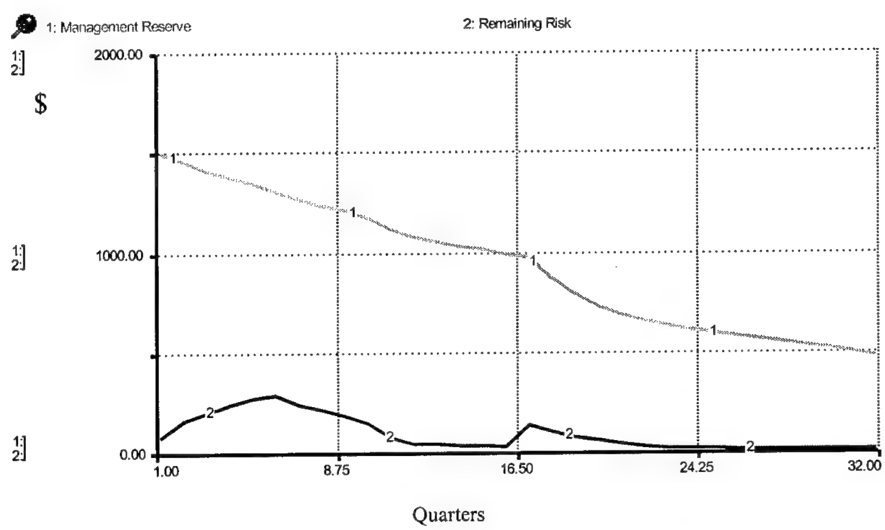


Figure 4.9 Management Reserve and Program Risk Behaviors with an Early Increase in Requirements

Late Requirement Changes:

The next scenario investigates the effect of late requirement changes in the last quarter of the development phase. The results are significantly different than early changes. Figure 4.10 illustrates increases in requirements at T=22, T=24, and T=26.

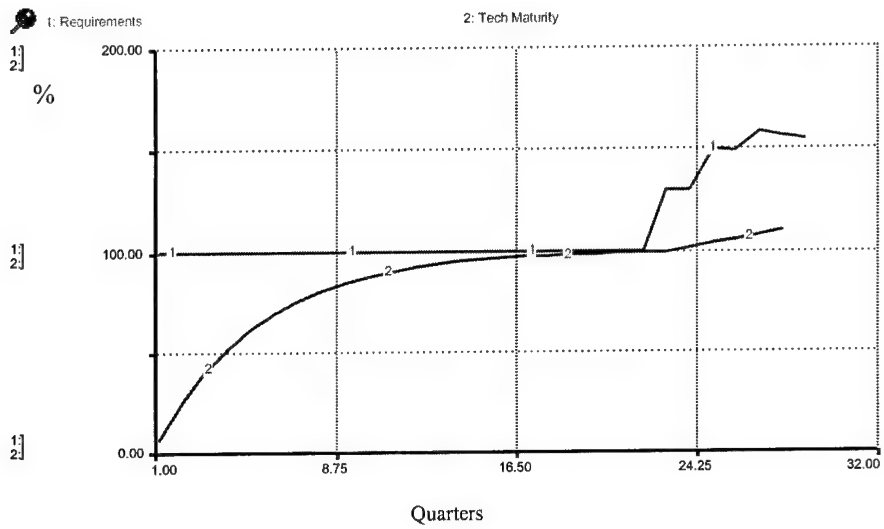


Figure 4.10 Tech Maturity and Requirement Behaviors with Late Increase in Requirements

Figure 4.11 goes unexecutable at T=25, when the risk exceeds the management reserve available. Notice that the program actually goes unexecutable after only a 25 % increase in the requirements.

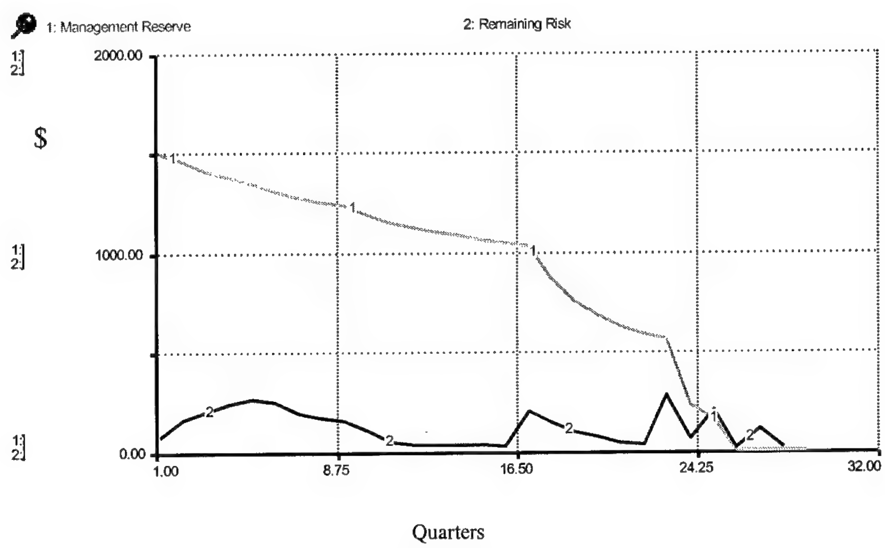


Figure 4.11 Management Reserve and Program Risk Behaviors with an Early Increase in Requirements

As expected, late increases in requirements have a grave impact on the performance of a program.

Sensitivity Analysis:

Once the extreme conditions were tested, we tested different policies and performed sensitivity analysis. To perform sensitivity analysis for this model, variables were individually manipulated while keeping all other variables constant. If the dynamic behavior of the model changed significantly, it was determined that the variable was a critical variable. The results of the sensitivity analysis determined that manpower selection for each of the three control variables, Manpower to Mitigate Risk, Manpower to Identify Risk, and Manpower to Answer Inquires, were significant factors in the model.

During the sensitivity analysis two manning scenarios were explored. The first looked at the ramifications of adequately manning the engineers who identify and mitigate risk in the beginning of the program, and then removing them at the midway point.

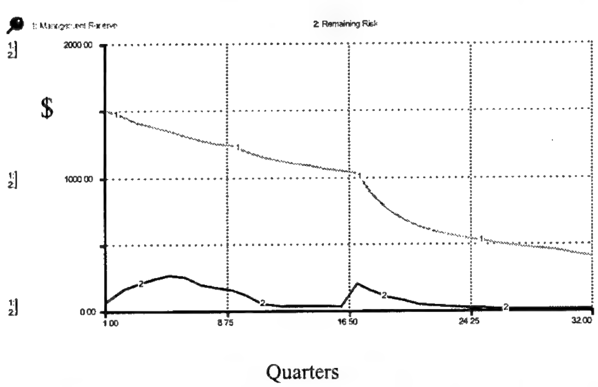


Figure 4.12 Management Reserve and Risk Performance with Adequate Manning

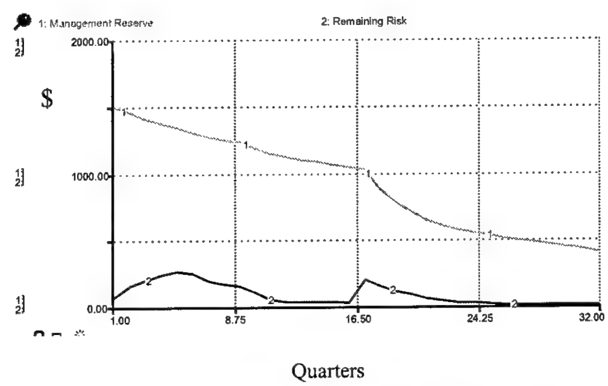


Figure 4.13 Management Reserve and Risk Performance with Adequate Manning time T=16 and no engineering manning T>16

The results demonstrated that the system is not sensitive to the removal of engineers who identify and mitigate risks after the midway point of a program development. This does not say to remove the engineering workforce completely, because engineers are still required to broker information to the stakeholders and to communicate with the contractor. However, the SPOs

might consider reallocating the engineering workforce to shift focus from risk identification and mitigation, to other areas.

The next scenario looked at the behavior of the system if the program has too few engineers at the beginning of the program, and then increases engineering staff at the midway point. Increasing engineering manpower did not improve the system risk performance. In fact, the program became unexecutable at T=20, see figure 4.15, versus the no engineering model which remains executable until T=24.

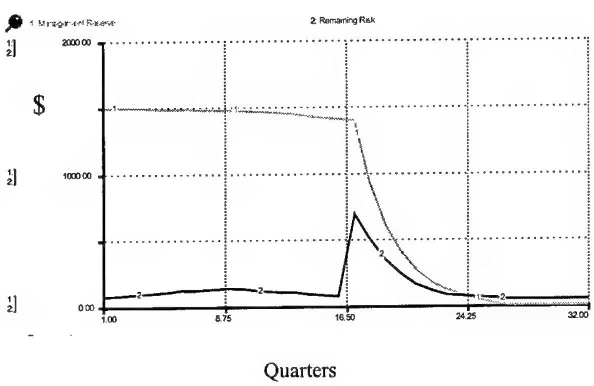


Figure 4.14 Management Reserve and Risk Performance with no manning

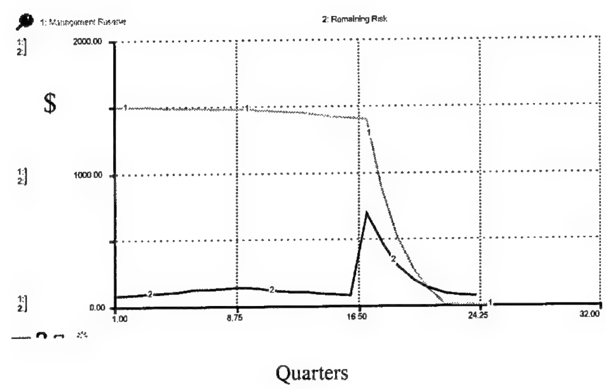


Figure 4.15 Management Reserve and Risk Performance with No Manning T=0 to T=16 and Adequate Manning T>16

This result is a bit counterintuitive. One might think that adding engineering support to a program would improve performance. However, the late addition of government engineering manpower would also increase the contractors engineering manpower. The added expense would causes the management reserve spending to increase at a faster rate.

Also, as mentioned in the section above, the system is sensitive to the changes in requirements. Early increases of requirements, does not affect the program. Late increases of requirements have grave consequences to system performance.

5. Conclusion/Recommendations

The exercise of generating the model has yielded a much greater understanding of the system. The systems experts have agreed that the basic structure of the current model is sound, and the boundary definition and the degree of aggregation is relatively close in order to provide insight into the system. Since system dynamics modeling is an iterative process, the next step would be to revisit the critical relationships and reach a consensus of the linear and nonlinear dynamic relationship through interviews and data gathering.

Thesis Objectives Revisited

The result of the thesis met or exceeded each objective.

The specific objectives were as follows:

1. ***Identify the structure of the system using traditional systems engineering tools such as Functional Analysis Systems Technique and Design Structure Matrix in concert with a system dynamics approach.*** The combination of the F.A.S.T. and the MCM with the final system dynamics model satisfied this objective.
2. ***Isolate the interactions and influence of the components and variables within the system.*** The MCM provided the structure and all of the critical influence and interrelationships were defined.
3. ***Describe the decision structure that determines the allocation of engineers to the SPOs.*** A system dynamics model was developed which describes and models the decision structure of the system.
4. ***Construct a mathematical model that represents the components, relationships, information flows, and decision policies of the system.*** A system of mathematical equations were developed and are outlined in Appendix D.

5. ***Develop a computerized model that can be used as a learning laboratory for policy analysis and development to optimize engineering support of a high-risk USAF development activity.*** An initial model was developed.
6. **Identify areas of sensitivity or critical issues in engineering manpower allocation.** Several areas of sensitivity were identified and are outlined in the previous chapter.

Customer Feedback

The system experts were very pleased with the results of the model. The largest USAF organic engineering organization has adopted the model as a baseline. The entire process enabled the system experts to see the system in a new light. The rigorous information gathering and the formal modeling process led USAF leadership ask the right questions. Each expert has expressed that there is value in pursuing a systems approach to solving the manpower problems over traditional empirical manpower models. They plan to further refine the existing model to better understand the critical dynamic relationships and will pursue better parameterization.

Summary

This thesis looked at the benefits of using System Engineering approach to system dynamics modeling. I believe there is mutual benefit. The system dynamics community can use the system engineering tools to investigate the system to be modeled. SE tools and processes can help the modeler to better understand the system to be modeled efficiently. The SE tools will also benefit the modeler in trying to define boundary and degree of aggregation.

In addition, I believe system dynamics simulation could be helpful to the SE community and consultants who are using SE process and tools above the shop floor. If correctly implemented, system dynamics could be used to enhance the SE tools and provide way for system engineers and managers to simulate management policy prior to implementation.

APPENDIX A: INTRODUCTION TO SYSTEM DYNAMICS

INTRODUCTION:

Causal diagrams and stocks and flow diagrams are the common tools used to provide a visual structure of a system. This thesis strictly used the stock and flow diagramming to model the system. The diagrams described in chapter three were generated using I-think, a computerized System Dynamics tool. This appendix contains a very brief introduction the system dynamics symbols used.

Stocks

Purpose: Stocks are accumulations. They collect whatever flows into and out of them.

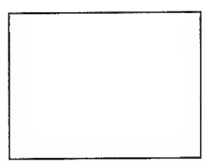


Figure A1 Stock

The default stock type is the Reservoir. Think of a Reservoir as a pool of water, or as an undifferentiated pile of "stuff." A Reservoir passively accumulates its inflows, minus its outflows. Any units which flow into a Reservoir will lose their individual identity. Reservoirs mix together all units into an undifferentiated mass as they accumulate.

Flows

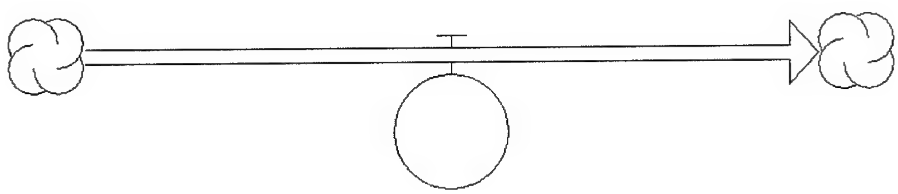


Figure A2 Flow

Purpose: The job of flows is to fill and drain accumulations. The unfilled arrow head on the flow pipe indicates the direction of the flow.

Uniflow vs. Biflow: Uniflow means that the flow will flow in one direction only. With uniflows, the flow volume will take on non-negative values only. On the other hand, biflows can

take on any value. If you specify a flow as a biflow, a second, shaded arrow head will appear on the flow to point the direction of negative flow. It is not possible to have a biflow connected to a conveyor, queue, or oven.

Unit Conversion: When the flow is conserved (i.e., it connects two stocks), the unit conversion check box is enabled. Unit conversion enables you to convert the units of measure for the flow, as material moves through the flow pipe. Unit conversion is useful in modeling processes such as assembly processes which transform raw materials into finished goods, or chemical processes which involve molecular transformations. When you specify a flow as unit converted, a shaded half-circle will appear in the flow regulator on the diagram to indicate that unit conversion is taking place.

Converters

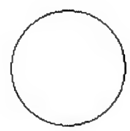


Figure A3 Converter

Purpose: The converter serves a utilitarian role in the software. It holds values for constants, defines external inputs to the model, calculates algebraic relationships, and serves as the repository for graphical functions. In general, it converts inputs into outputs. Hence, the name converter.

Connectors

Figure A3 Connector

Purpose: As its name suggests, the job of the connector is to connect model elements. Legal connector linkages are illustrated in the illustration.

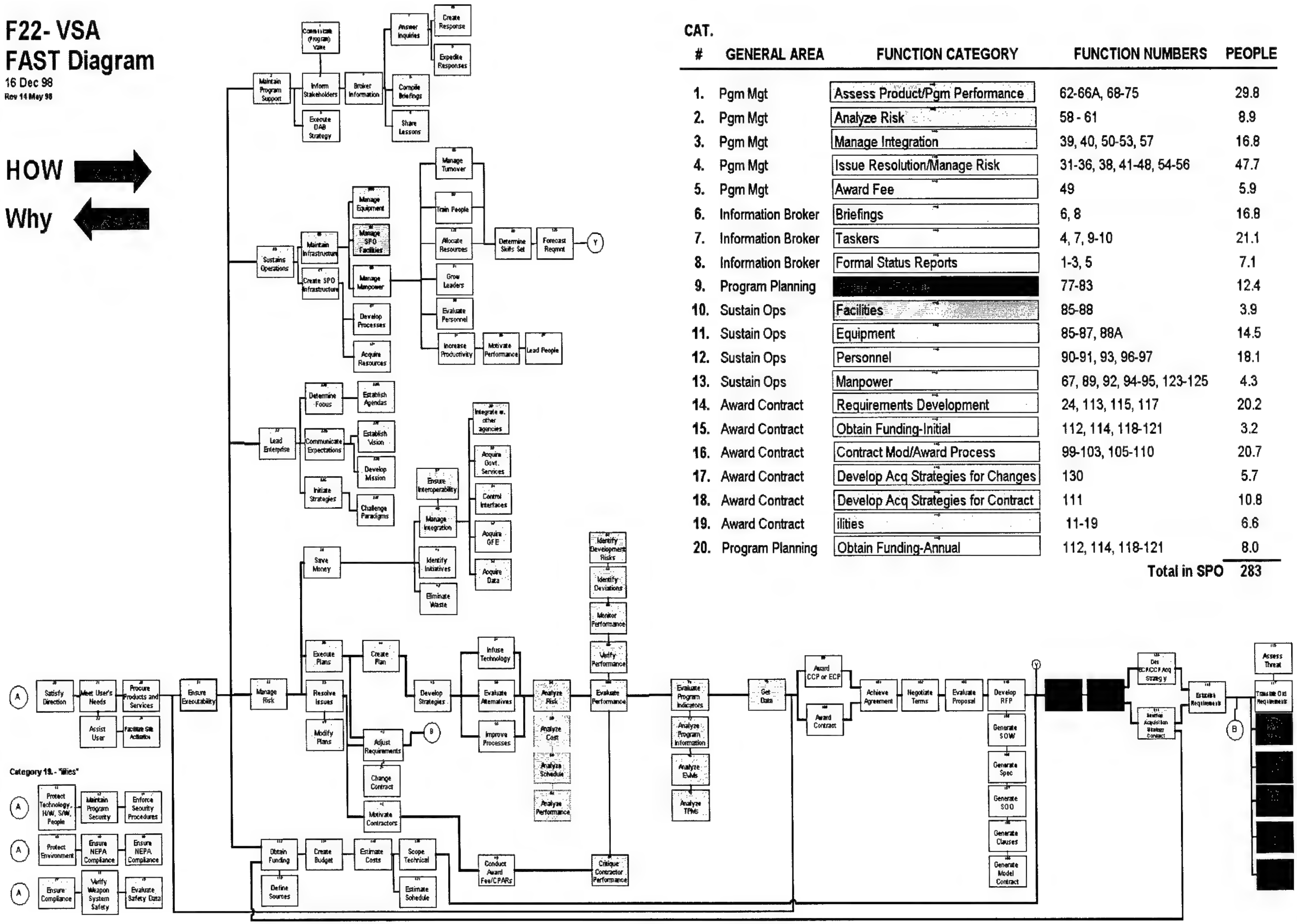
You can never drag a connector into a stock. The only way to change the magnitude of a stock, is through a flow.

**The information was taken from I-Think® Online Help Manual

APPENDIX B: F-22 F.A.S.T. DIAGRAM

F22- VSA FAST Diagram

16 Dec 98
Rev 14 May 98



CAT. #	GENERAL AREA	FUNCTION CATEGORY	FUNCTION NUMBERS	PEOPLE
1.	Pgm Mgt	Assess Product/Pgm Performance	62-66A, 68-75	29.8
2.	Pgm Mgt	Analyze Risk	58 - 61	8.9
3.	Pgm Mgt	Manage Integration	39, 40, 50-53, 57	16.8
4.	Pgm Mgt	Issue Resolution/Manage Risk	31-36, 38, 41-48, 54-56	47.7
5.	Pgm Mgt	Award Fee	49	5.9
6.	Information Broker	Briefings	6, 8	16.8
7.	Information Broker	Taskers	4, 7, 9-10	21.1
8.	Information Broker	Formal Status Reports	1-3, 5	7.1
9.	Program Planning	[redacted]	77-83	12.4
10.	Sustain Ops	Facilities	85-88	3.9
11.	Sustain Ops	Equipment	85-87, 88A	14.5
12.	Sustain Ops	Personnel	90-91, 93, 96-97	18.1
13.	Sustain Ops	Manpower	67, 89, 92, 94-95, 123-125	4.3
14.	Award Contract	Requirements Development	24, 113, 115, 117	20.2
15.	Award Contract	Obtain Funding-Initial	112, 114, 118-121	3.2
16.	Award Contract	Contract Mod/Award Process	99-103, 105-110	20.7
17.	Award Contract	Develop Acq Strategies for Changes	130	5.7
18.	Award Contract	Develop Acq Strategies for Contract	111	10.8
19.	Award Contract	ilities	11-19	6.6
20.	Program Planning	Obtain Funding-Annual	112, 114, 118-121	8.0
			Total in SPO	283

F.A.S.T. diagram Colors

The F-22 SPO F.A.S.T. diagram shows its primary functions are the following: maintain support, sustain operations, lead the enterprise (AF – contractor combination), manage risk, and obtain funding. Each of these primary functions is followed by supporting functions that are ordered using the how-why logic explained earlier.

The colors on the F.A.S.T. diagram were added later when the F-22 teams decided to allocate people to the F.A.S.T. It was discovered that this allocation could not easily be done by function. Therefore, the diagram was subdivided by color and people were allocated by everyday activities even when they crossed functional areas.

The F-22 SPO's diagram was created in one day. Its form is that of the technical F.A.S.T. diagram. The information provided by the F.A.S.T. model was used in the information-gathering phase during the development of the organization's system dynamics model of the engineering function. It was used in concert with other value analysis tools to improve the management function of the SPO.

APPENDIX C: MANAGEMENT CAUSAL MATRIX

Management Causal Matrix

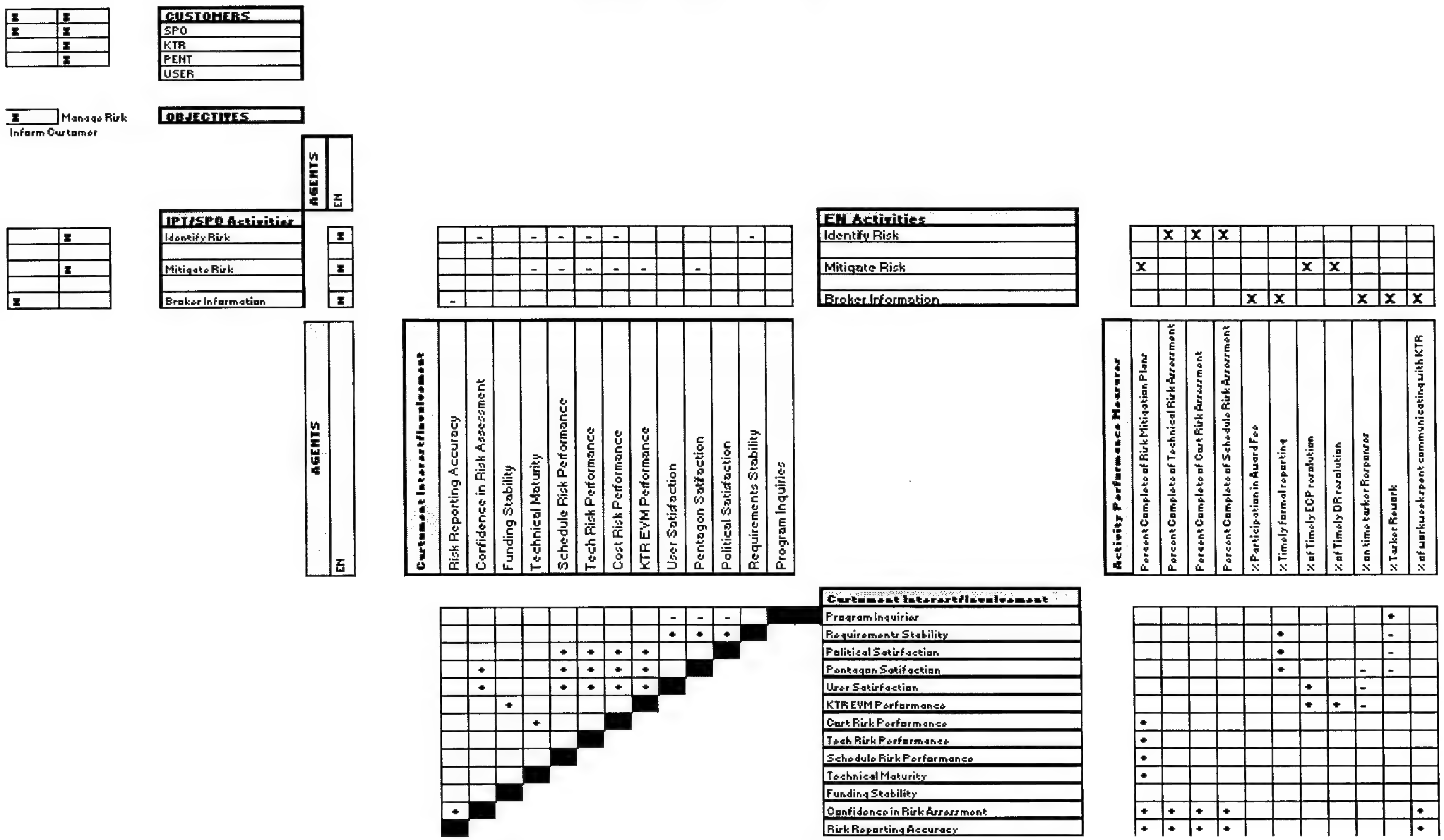


Figure C.1 Management Causal Matrix of Government Engineering Support for a Development Program

APPENDIX D: MODEL EQUATIONS

Cost/Schedule Performance Sector

Management_Reserve(t) = Management_Reserve(t - dt) + (MR_In - MR_Out) * dt

INIT Management_Reserve = Initial_Management_Reserve

INFLOWS:

MR_In = New_MR

OUTFLOWS:

MR_Out =

(Cost_to_Mitigate_Risk*Mitigated_Risk_Out+MR_Applied_to_Risk+MR_Risk_Out*Cost_to_

Mitigate_Risk)*(1/Units_of_Risk_Per_unit_of_MR)+MR_Reduction_Due_to_Overmanning

Program_Schedule(t) = Program_Schedule(t - dt) + (Schedule_IN - Schedule_Out) * dt

INIT Program_Schedule = Initial_Milestone_III_Decision

INFLOWS:

Schedule_IN = New_Requirements*Time_Factor_New_Rqmt_and_Schedule_2

+Schedule_and_risk_time_factor_In*Reported_Risk_In

OUTFLOWS:

Schedule_Out = Mitigated_Risk_Out*Schedule_and_risk_time_factor_Out+Reqt's_Out

*Time_Factor_New_Rqmt_and_Schedule

BAC = Defined_BAC

Business_Performance =

(Defined_EAC+(Reported_Program_Risk/Units_of_Risk_Per_unit_of_MR))

/(Defined_BAC+Management_Reserve)

EAC = Defined_EAC

Initial_Management_Reserve = 0

MR_Reduction_Due_to_Overmanning = If

(Difference_in_IndentifyRisk_Manpower+Difference_in_Inquiries_Manpower+Difference_in_Mitigate_Risk_Manpower)>0 then

(Difference_in_IndentifyRisk_Manpower+Difference_in_Inquiries_Manpower+Difference_in_Mitigate_Risk_Manpower)/Units_of_Risk_Per_unit_of_MR else 0

Units_of_Risk_Per_unit_of_MR = 10

Cost_to_Mitigate_Risk = GRAPH(TIME) (0.00, 1.00), (4.00, 1.00), (8.00, 1.00), (12.0, 3.00), (16.0, 3.55), (20.0, 4.00), (24.0, 6.20), (28.0, 15.6), (32.0, 39.6)

Defined_BAC = GRAPH(Time) (1.00, 0.00), (4.10, 0.00), (7.20, 0.00), (10.3, 0.00), (13.4, 0.00), (16.5, 0.00), (19.6, 0.00), (22.7, 0.00), (25.8, 0.00), (28.9, 0.00), (32.0, 0.00)

Defined_EAC = GRAPH(TIME) (1.00, 0.00), (4.10, 0.00), (7.20, 0.00), (10.3, 0.00), (13.4, 0.00), (16.5, 0.00), (19.6, 0.00), (22.7, 0.00), (25.8, 0.00), (28.9, 0.00), (32.0, 0.00)

New_MR = GRAPH(TIme) (0.00, 0.00), (4.00, 0.00), (8.00, 0.00), (12.0, 0.00), (16.0, 0.00), (20.0, 0.00), (24.0, 0.00), (28.0, 0.00), (32.0, 0.00)

Schedule_and_risk_time_factor_In = GRAPH(Time) (0.00, 0.00), (4.00, 0.00), (8.00, 0.0005), (12.0, 0.0035), (16.0, 0.009), (20.0, 0.0155), (24.0, 0.026), (28.0, 0.0495), (32.0, 0.062)

Schedule_and_risk_time_factor_Out = GRAPH(Time) (0.00, 0.00), (4.00, 0.00), (8.00, 0.00), (12.0, 0.00), (16.0, 0.0413), (20.0, 0.0362), (24.0, 0.0138), (28.0, 0.0075), (32.0, 0.005)

Time_Factor_New_Rqmt_and_Schedule = GRAPH(time) (1.00, 0.76), (11.3, 0.37), (21.7, 0.2), (32.0, 0.01)

Time_Factor_New_Rqmt_and_Schedule_2 = GRAPH(time) (1.00, 0.03), (11.3, 0.39), (21.7, 1.61), (32.0, 1.99)

Inquiries Sector Inquiries(t) = Inquiries(t - dt) + (Inquiries_In - Inquiries_Out) * dt

INIT Inquiries = 0

TRANSIT TIME = 2

INFLOW LIMIT = INF

CAPACITY = INF

INFLOWS:

Inquiries_In =

Planned_Inquiries+Inquiries_Due_to_Program_Performance+Inquiries_Due_to_Customer_Satis

faction

OUTFLOWS:

Inquiries_Out = CONVEYOR OUTFLOW

Answer_Inquiries_Factor =

(Available_Manpower_to_Answer_Inquiries/Required_Effort_Inquiries)

Difference_in_Inquiries_Manpower = Available_Manpower_to_Answer_Inquiries-

Required_Effort_Inquiries

Inquiries_Due_to_Customer_Satisfaction = If Answer_Inquiries_Factor>1.25

then(Inquiries_Out*.5) else 0

Inquiries_Due_to_Program_Performance = GRAPH(Business_Performance)

(0.00, 4.00), (10.0, 6.00), (20.0, 7.00), (30.0, 8.00), (40.0, 9.00), (50.0, 10.0), (60.0, 10.0), (70.0, 10.0), (80.0, 10.0), (90.0, 10.0), (100, 10.0)

Planned_Inquiries = GRAPH(time)

(0.00, 2.50), (4.00, 2.50), (8.00, 2.50), (12.0, 2.50), (16.0, 5.00), (20.0, 10.0), (24.0, 22.5), (28.0, 25.0), (32.0, 25.0)

Required_Effort_Inquiries = GRAPH(Inquiries)

(0.00, 10.0), (10.0, 42.0), (20.0, 99.0)

Manpower Sector

Available_Manpower_to_Answer_Inquiries = 10

Available_Manpower_to_Identify_Risk = 100

Available_Manpower_to_Mitigate_Risk = 10

EN_Manpower =

Available_Manpower_to_Answer_Inquiries+Available_Manpower_to_Identify_Risk+Available_Manpower_to_Mitigate_Risk

Program Maturity Sector

Initial_Maturity = 5

Initial_Milestone_III_Decision = 32

Projected_Capability = 95

Tech_Maturity =

SMTH1(Requirements,(Program_Schedule/Initial_Milestone_III_Decision*5),Initial_Maturity)

Requirements Sector

Requirements(t) = Requirements(t - dt) + (Reqt's_In - Reqt's_Out) * dt

INIT Requirements = 100

INFLOWS:

Reqt's_In = New_Requirements

OUTFLOWS:

Reqt's_Out = GRAPH(Users_Willingness_to_Relieve_Requirements)

(0.00, 0.00), (0.1, 0.1), (0.2, 0.35), (0.3, 1.15), (0.4, 1.65), (0.5, 2.20), (0.6, 3.55), (0.7, 4.75), (0.8, 6.85), (0.9, 8.50), (1, 10.0)

Performance_Reqts_Gap = (Tech_Maturity/Requirements)*100

New_Requirements = GRAPH(TIME)

(1.00, 0.00), (2.00, 0.00), (3.00, 0.00), (4.00, 0.00), (5.00, 0.00), (6.00, 0.00), (7.00, 0.00), (8.00, 0.00), (9.00, 10.0), (10.0, 0.00), (11.0, 0.00), (12.0, 0.00), (13.0, 0.00), (14.0, 0.00), (15.0, 0.00), (16.0, 0.00), (17.0, 0.00), (18.0, 0.00), (19.0, 0.00), (20.0, 0.00), (21.0, 0.00), (22.0, 0.00), (23.0, 0.00), (24.0, 0.00), (25.0, 0.00), (26.0, 0.00), (27.0, 0.00), (28.0, 0.00), (29.0, 0.00), (30.0, 0.00), (31.0, 0.00), (32.0, 0.00)

Users_Willingness_to_Relieve_Requirements = GRAPH(Business_Performance)

(0.00, 0.00), (0.2, 0.00), (0.4, 0.00), (0.6, 0.00), (0.8, 0.00), (1.00, 0.00), (1.20, 0.1), (1.40, 0.39), (1.60, 0.56), (1.80, 0.8), (2.00, 0.8)

Risk Sector

Management_Challenge(t) = Management_Challenge(t - dt) + (Management_Challenge_In - Mitigated_Risk_Out - Unmitigatable_Risk) * dt

INIT Management_Challenge = Initial_Identified_Program_Risk*.3

INFLOWS:

Management_Challenge_In = Risk_IN*Selected_Management_Challenge

OUTFLOWS:

Mitigated_Risk_Out = Management_Challenge*Mitigate_Risk_Factor

Unmitigatable_Risk = Management_Challenge*Unmitigateable_risk_factor

Reported_Program_Risk(t) = Reported_Program_Risk(t - dt) + (Reported_Risk_In - MR_Risk_Out - Requirements_Relief_Risk_Out) * dt

INIT Reported_Program_Risk = Initial_Identified_Program_Risk

INFLOWS:

Reported_Risk_In = (1-Selected_Management_Challenge)*Risk_IN+.3*Mitigated_Risk_Out

OUTFLOWS:

MR_Risk_Out =

delay(Reported_Risk_In,1)+MR_Applied_to_Risk*(Units_of_Risk_Per_unit_of_MR)

Requirements_Relief_Risk_Out = Risk_Decrease_Due_to_Requirements_Relief

Units_of_Risk_to_be_Discovered(t) = Units_of_Risk_to_be_Discovered(t - dt) + (Discovery_Profile - Discovered_Risk) * dt

INIT Units_of_Risk_to_be_Discovered = 0

INFLOWS:

Discovery_Profile = GRAPH(TIME)

(1.00, 80.0), (5.43, 67.0), (9.86, 58.5), (14.3, 27.5), (18.7, 6.00), (23.1, 2.00), (27.6, 0.00), (32.0, 0.00)

OUTFLOWS:

Discovered_Risk = if time<16 then Discovery_Profile*Rate_of_Risk_Discovery else Units_of_Risk_to_be_Discovered*.5

Confidence_in_Risk_Assessment =

Available_Manpower_to_Identify_Risk/Required_Effort_to_Indentify_Risk

Confidence_in_Risk_Mitigation =

(Available_Manpower_to_Mitigate_Risk/Manpower_Required_to_Mitigate_Risk)

Difference_in_IndentifyRisk_Manpower = Available_Manpower_to_Identify_Risk-

Required_Effort_to_Indentify_Risk

Difference_in_Mitigate_Risk_Manpower = Available_Manpower_to_Mitigate_Risk-

Manpower_Required_to_Mitigate_Risk

Initial_Identified_Program_Risk = 0

MR_Applied_to_Risk = 0

Remaining_Risk = (Reported_Program_Risk+Management_Challenge)

Risk_Decrease_Due_to_Requirements_Relief =

1/New_Requirement_Risk_In_Factor*Reqt's_Out

Risk_IN = Discovered_Risk+Unmitigatable_Risk+ Risk_Increase_Due_to_New_Requirements

Risk_Increase_Due_to_New_Requirements =

New_Requirements*New_Requirement_Risk_In_Factor

Manpower_Required_to_Mitigate_Risk = GRAPH(Management_Challenge)

(0.00, 7.00), (10.0, 7.50), (20.0, 10.0), (30.0, 12.0), (40.0, 14.5), (50.0, 19.0), (60.0, 24.0), (70.0, 33.0), (80.0, 42.0), (90.0, 55.5), (100, 83.0)

Mitigate_Risk_Factor = GRAPH(Confidence_in_Risk_Mitigation)

(0.00, 0.00), (0.5, 0.285), (1.00, 0.535), (1.50, 0.69), (2.00, 0.76)

New_Requirement_Risk_In_Factor = GRAPH(TIME)

(1.00, 0.1), (4.10, 0.1), (7.20, 0.1), (10.3, 0.1), (13.4, 0.95), (16.5, 4.05), (19.6, 7.55), (22.7, 8.95), (25.8, 9.40), (28.9, 9.90), (32.0, 10.0)

Rate_of_Risk_Discovery = GRAPH(Confidence_in_Risk_Assessment)

(0.00, 0.125), (0.2, 0.165), (0.4, 0.22), (0.6, 0.285), (0.8, 0.35), (1.00, 0.455), (1.20, 0.555), (1.40, 0.645), (1.60, 0.73), (1.80, 0.855), (2.00, 1.00)

Required_Effort_to_Indentify_Risk = GRAPH(Performance_Reqts_Gap)

(0.00, 3.40), (10.0, 4.00), (20.0, 4.00), (30.0, 6.00), (40.0, 14.8), (50.0, 28.4), (60.0, 42.0), (70.0, 55.2), (80.0, 63.2), (90.0, 70.0), (100, 72.8)

Selected_Management_Challenge = GRAPH(TIME)

(0.00, 0.5), (8.00, 0.395), (16.0, 0.2), (24.0, 0.0525), (32.0, 0.0525)

Unmitigateable_risk_factor = GRAPH(TIME)

(0.00, 0.00), (4.00, 0.00), (8.00, 0.25), (12.0, 0.5), (16.0, 0.7), (20.0, 0.8), (24.0, 0.9), (28.0, 1.00), (32.0, 1.00)

BIBLIOGRAPHY

Blanchard, B.S. and Fabrycky, W. J. Systems Engineering and Analysis. Prentice Hall: Upper Saddle River NJ, 1998.

Elder, J.E. and Mixon, M.B. 1976. " The Application of System Dynamics to a Management Model of Aeronautical Systems Division." Unpublished master's thesis. LSSR 22-78A, AFIT/LS, Wright-Patterson AFB OH, June 1976

Fowler T. Value Analysis in Design. Van Nostrand Reinhold: New York, 1990.

Ford, D. and Sterman, J. "Expert knowledge elicitation for improving mental and formal models," System Dynamics Review 14(4), 309-340, 1998a.

Forrester, J. W. Industrial Dynamics. The MIT Press: Cambridge MA, 1961.

Druyun, D. Principal Deputy Assistant Secretary of the Air Force for Acquisition And Management, Washington DC, Personal Correspondence and Report, 6 Nov 1995.

Gonzalez, J.F. and Sarama, T.J. "Lean Methods Above the Manufacturing Floor in a Aerospace Military Program." Unpublished master's thesis. Massachusetts Institute of Technology, Cambridge MA, 1999.

Hill, J. and Warfield, J. "Unified Program Planning," IEEE Transactions of Systems, Man, and Cybernetics 2(5), 610-623, 1972.

Kaffenberger, S.A. and Martin, D. P. "A Dynamic Policy Model of the Defense Systems Acquisitions Process." Unpublished master's thesis. LSSR 1-79A, AFIT/LS, Wright-Patterson AFB OH, 1979.

Lawson, D. and Osterhaus, D.L. "A Conceptual Model of the Department of Defense Major System Acquisition Process." Unpublished master's thesis. LSSR 1-79A. AFIT/LS, Wright-Patterson AFB OH, 1978.

Sterman, J. Business Dynamics. McGraw-Hill: Boston MA, 2000.

Yates, R. W. Commander, Air Force Systems Command, Andrews Air Force Base, MD, Personal Correspondence, 24 June 1991.

VITA

Lieutenant Jason E. Bartolomei was born in Janesville, Wisconsin. He graduated from Alamo Heights High School in San Antonio, Texas in June 1993. He entered undergraduate studies at Marquette University in Milwaukee, Wisconsin where he graduated with a Bachelor of Science degree in Mechanical Engineering in August 1997. He was commissioned through AFROTC at Marquette University.

His first assignment was at Wright-Patterson AFB, Ohio as a developmental engineer. While at Wright-Patterson, he served as an engineer and program manager for the F-22 System Program Office and as the Executive Officer for the Aeronautical Systems Center's Engineering Directorate. Upon graduation, he will be assigned to the United States Air Force Academy as an instructor of engineering mechanics.

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
03-09-2001	Master's Thesis	Jun 1999 – Sep 2001

4. TITLE AND SUBTITLE	**5a. CONTRACT NUMBER**
THE USE OF SYSTEMS ENGINEERING PROCESSES AND TOOLS TO DEVELOP A SYSTEM DYNAMIC SIMULATION MODEL OF ENGINEERING SUPPORT DURING THE DEVELOPMENT PHASE OF AN ACQUISITION PROGRAM	**5b. GRANT NUMBER**
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	**5d. PROJECT NUMBER**
Bartolomei, Jason, E., First Lieutenant, USAF	**5e. TASK NUMBER**
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/ENY) 2950 P Street, Building 640 WPAFB OH 45433-7765	AFIT/GSE/ENY/01M-01
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**
ASC/EN Attn: Mr. Jon S. Ogg 2430 Loop Road West WPAFB OH 45433-7107	**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Due to the increase of system complexity and the existing draw down of manpower allocations, today's acquisitions environment desperately needs a systems approach to decision making. Many studies have been performed to model the entire government acquisition environment. Due to the high degree of aggregation, front line decision-makers have had no use for the information these models provide.

This research focuses on the Air Force's largest functional support element in aircraft systems development, engineering. I will only consider one phase of the government acquisition cycle the Engineering, Manufacturing, and Development (EMD). This is the development cycle, which begins with initial contract award (Milestone II), through the production approval (Milestone III). The structure of this model will be a building block to help USAF leadership in the determination of required engineering skill-set and manpower to perform activities which can meet short term requirements while minimizing the intrinsic cost, schedule, and performance risks associated system development. The simulation model will be used by USAF leadership as an alternative decision making tool for manpower allocations for government organic engineering workforce during an eight year development effort.

In addition, this study investigates the benefit of using system engineering tools and processes, like Functional Allocation (FAST) and Quality Functional Deployment, to improve the process for generating system dynamics simulation models.

For years, the systems engineering field has developed tools to graphically represent complex system structure. Graphical representations allow individuals and teams to visually identify interrelationships and dependencies within a system. Academic research and the successful implementation of these tools within the industrial communities validate the utility of these tools. These tools include Unified Program Planning, Quality Functional Deployment, House of Quality, and Design Structure Matrix.

This thesis presents a new tool, Management Causal Matrix, for system dynamics modeling community. The matrix is very similar to the more traditional systems engineering tools, yet has been customized for the systems dynamacist to highlight system interdependencies and organize the causal structure for a management system

A system dynamic simulator was developed to examine government engineering resource allocation during the development phase of an acquisition program. By using systems engineering approach, the scope of the previously poorly understood system was efficiently determined and a dynamic model was produced.

15. SUBJECT TERMS
System Dynamics, System Engineering, Functional Analysis System Technique, Acquisitions, Matrix, Complex system design

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Lieutenant Colonel E. P. Smith
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UU	95	**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U			(937) 255-6565, ext 4318

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

Form Approved
OMB No. 074-0188